POSTFLIGHT REPORTER PROGRAM

project
mercury

prepared for NASA

MC 110

project
mercury

POSTFLIGHT REPORTER PROGRAM

prepared for

National Aeronautics and Space Administration

Contract No. NAS 1-430

1 july 1961

International Business Machines Corporation

in association with

WESTERN ELECTRIC COMPANY, INC.

LIST OF EFFECTIVE PAGES

* The asterisk indicates pages changed, added, or deleted by the current change.

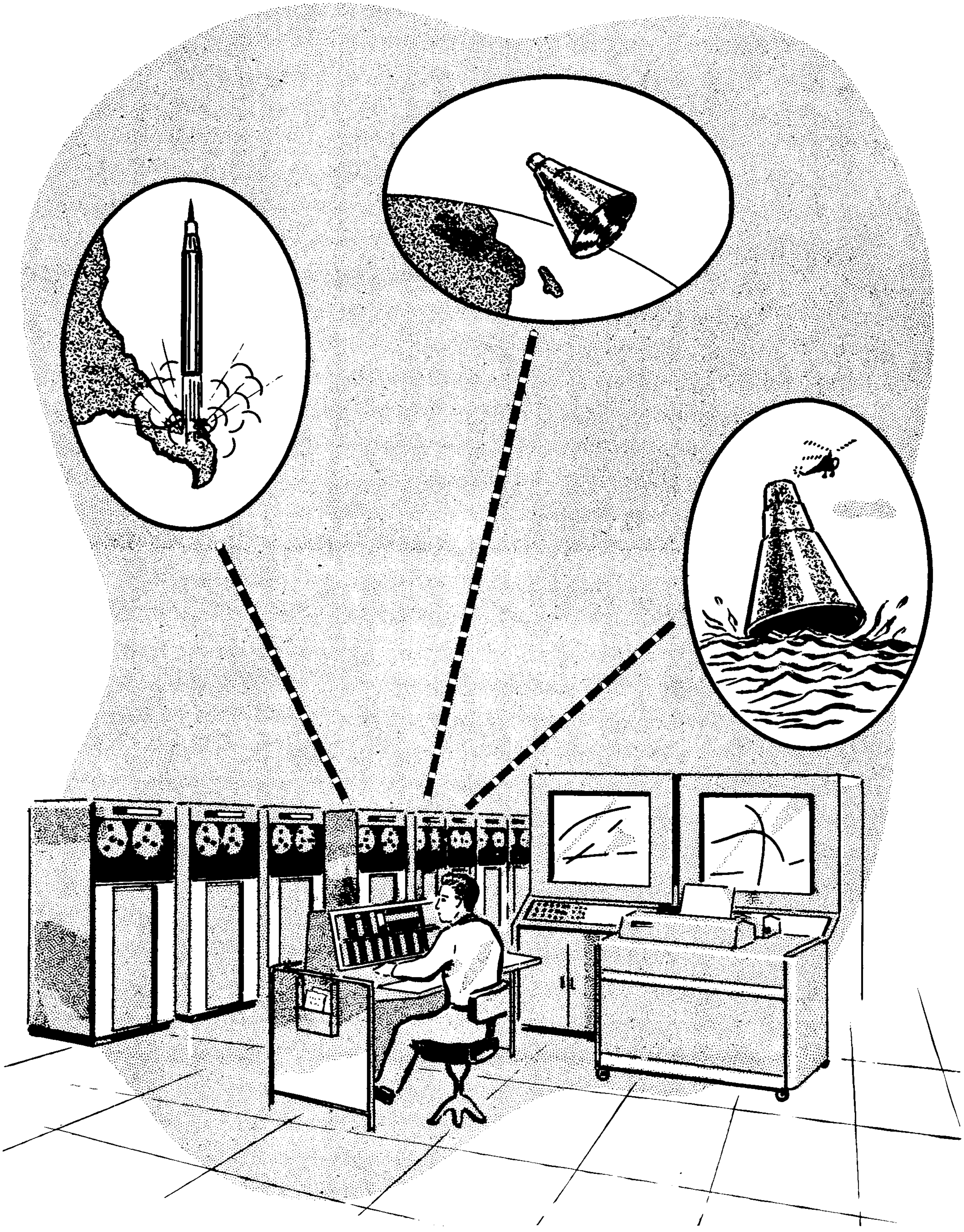


TABLE OF CONTENTS

TABLE OF CONTENTS (Cont'd)

LIST OF ILLUSTRATIONS

SECTION 1
INTRODUCTION

The Postflight Reporter Program produces summary reports of pertinent recorded data on Mercury missions for purposes of postflight analyses by Project Mercury personnel.

Whenever the Mercury Program System is put into operation—whether for simulated or real time missions—a log tape is generated which records all inputs and outputs between the Goddard Communications Center, the Mercury Control Center at Cape Canaveral and the world-wide tracking network. The information recorded on the log tape provides the basis for the summary reports of the Postflight Reporter Program. These summaries are then made available to Goddard, Mercury Control Center and Langley Field personnel for analysis.

1.1 NATURE OF POST FLIGHT REPORTS.

As specified in the NASA-Space Task Group Memorandum (December 14, 1960) the postflight reports are of two kinds.

a) The first is a Quick Look Report. Sufficient data for initial correlation and qualitative summary of test results are transmitted from Goddard Space Flight Center to the Mercury Control Center at Cape Canaveral within 24 hours after the Goddard computer is released from the mission. The data are transmitted over the teletype data link, according to the formats specified in the NASA-STG memorandum (see appendix C). There is also provision for listing of special data to be prepared at Goddard and handcarried to Cape Canaveral.

b) The second is a more detailed Three Day Report. This is delivered to the NASA-Space Task Group at Langley Field, Virginia, within three days after the flight.

These postflight reports may cover either an Atlas or a Redstone mission. Each report—whether quick-look, or detailed in nature—provides specific data for each applicable phase of the flight—Launch, Abort, Orbit and Re-entry. The Orbit and Re-entry phases would not be included for a Redstone mission.

Special Characteristics: There are some characteristics that differentiate the Postflight Reporter Program from the Mercury Program System. The Reporter is a self-contained program written almost entirely in 32K 709/7090

FORTRAN language; three major sub-routines are coded in FAP. The Mercury System, with the exception of some external routines, is written in Share Operating System (SOS) language.

The program uses shared storage locations, COMMON, for all input and output. From COMMON, output may be either read onto tape for off-line printing and teletype conversion, or printed out on-line. The program does not operate in a real time environment, and there is no core storage problem. The entire program including COMMON uses approximately 16,000 decimal locations. This is less than half of core storage.

Finally, in addition to utilizing data directly available on the Mercury System log tape, the Reporter program also recalculates certain parameters, and it calculates parameters which either are never calculated by the operational program (for example, aerodynamic parameters) or which are calculated but never recorded (for example, longitude of the node).

1.2 SCOPE OF THE MANUAL

This volume discusses the broad logic used in the Postflight Reporter Program. It describes the control program, Monitor, and indicates where the subprograms fit into it. This discussion is included with the description of Monitor in Section 2. The subprograms used with Monitor are described in Sections 3 through 6, and program operating procedures are detailed in Section 7. The general organization of the Postflight Reporter Program is shown in Fig. 1-1. Finally, there are three appendices—Postflight Reporter Symbolic Designations (Appendix A), Coordinate Conversion Systems (Appendix B), and Report Data Formats (Appendix C).

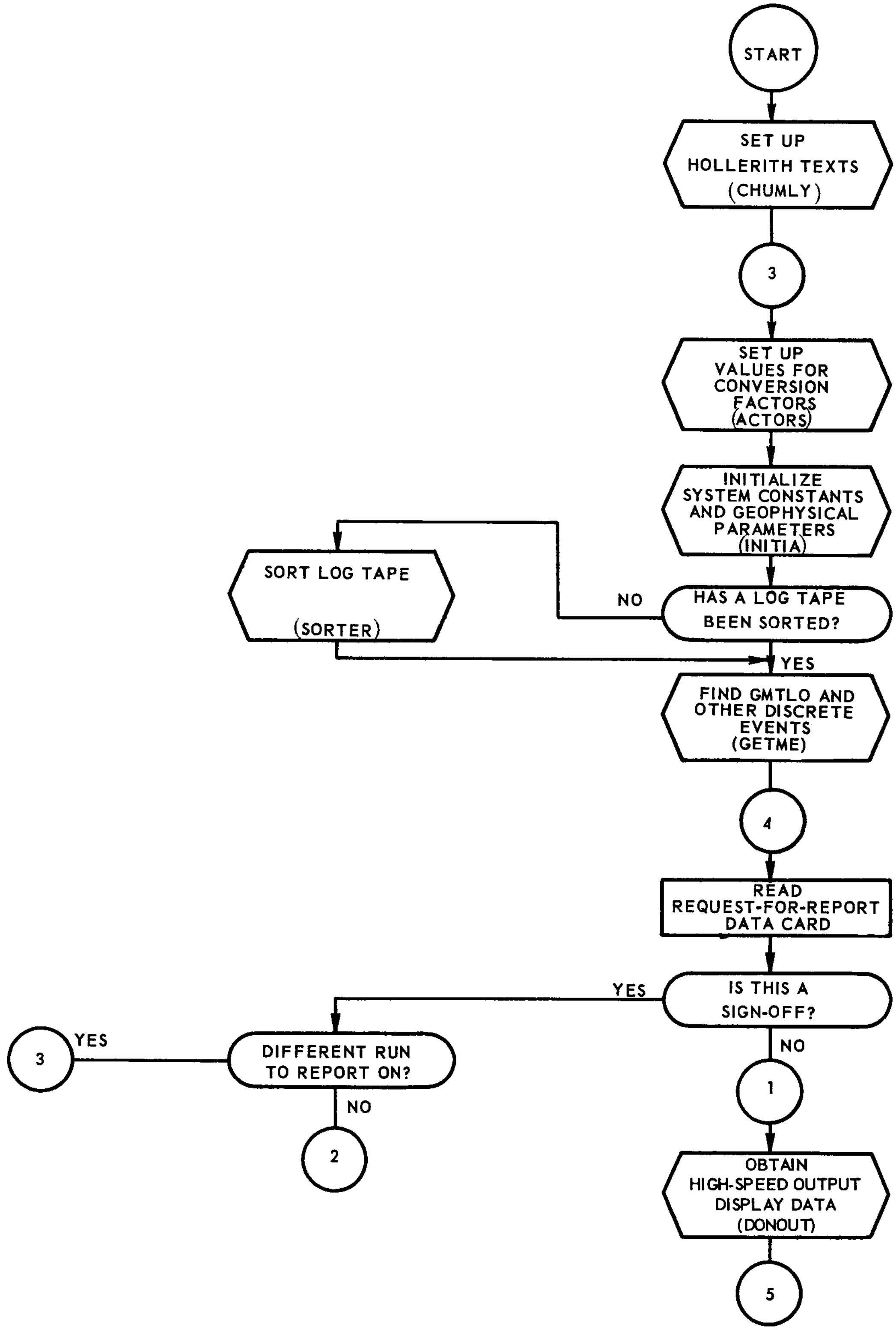


FIGURE 1-1. POSTFLIGHT REPORTER PROGRAM (GENERAL FLOW CHART) *(Sheet 1 of 2)*

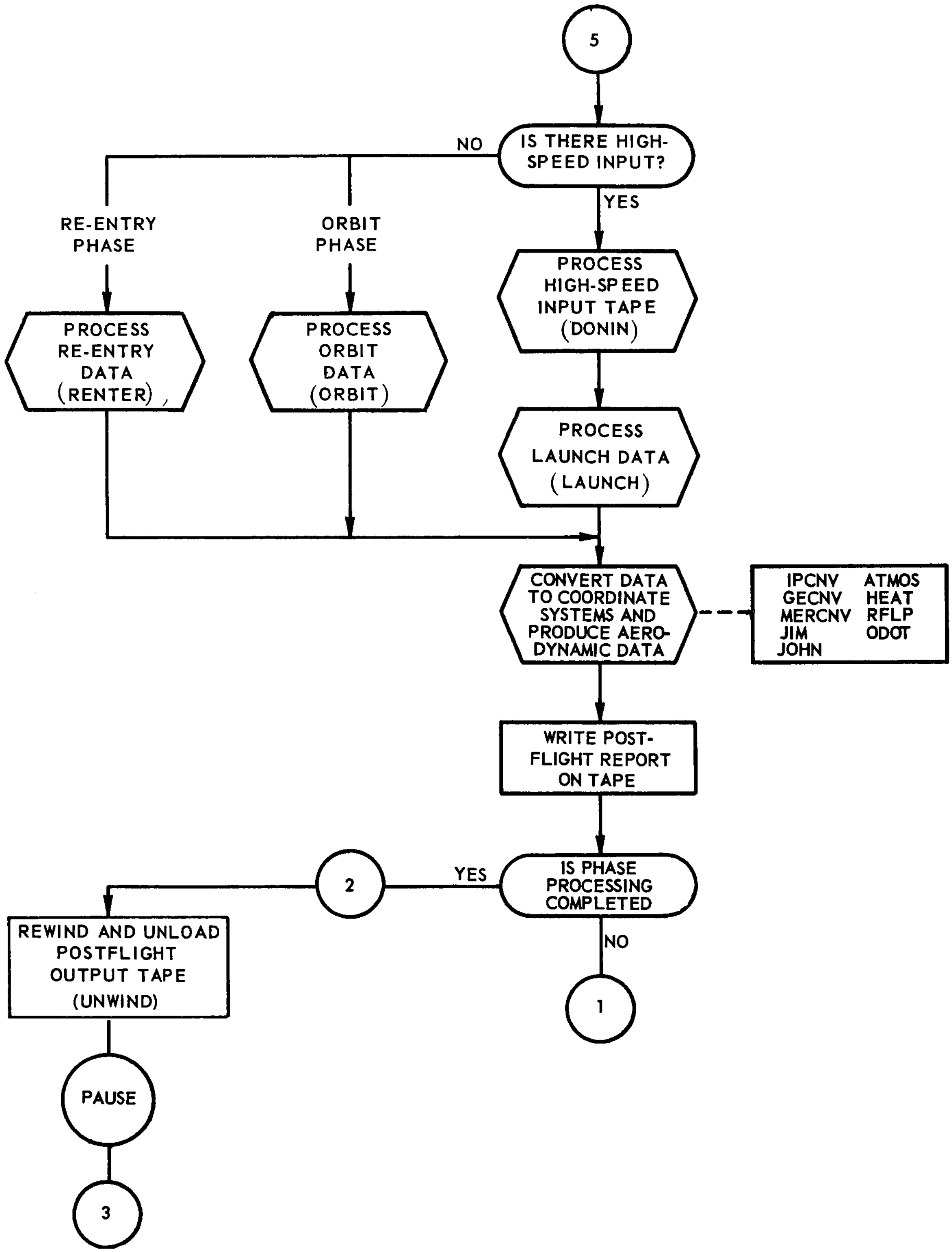


FIGURE 1.1. POSTFLIGHT REPORTER PROGRAM (GENERAL FLOW CHART) *(Sheet 2 of 2)*

SECTION 2
MONITOR PROGRAM

The Monitor program is a control program developed specifically for postflight reporter purposes. It is a separate program in no way related to the Mercury Program System Monitor. It makes logical decisions and directs the functioning of the various subroutines. Its operations directly produce the postflight reports. The Postflight Monitor supervises the search for output, from which corresponding input is derived. The program arrives at causes from results, which is the reverse of the procedure employed in the Mercury Program System.

Monitor relies on external and internal controls. Internally, it uses COMMON to facilitate the exchange of data among its subroutines. COMMON creates storage for the general interchange of information. Externally, Monitor requires the use of control data cards, sense switches and entry keys to exercise the various options of the program.

The flow chart for the Monitor program is shown in Figure 2-1.

2.1 INPUT REQUIREMENTS

Ten major subprograms are inputs to the Monitor program. They are:

CHUMLY -- sets up BCD output blocks in core

ACTORS -- sets conversion constants into core

INITIA -- initializes system constants and geophysical parameters

SORTER -- sorts the log tape into high-speed input and high-speed output data and determines discrete event timing

GETME -- supplies times of discrete events

DONOUT -- obtains high-speed output data

DONIN -- obtains high-speed input data

LAUNCH -- calculates launch and abort parameters from high-speed input data

ORBIT -- calculates orbit parameters from orbit display output data

RENTER -- calculates abort and re-entry parameters from the re-entry display output data

These major inputs use a number of subordinate programs which are discussed later in this manual in the sections dealing with the major subprograms (Sections 3 through 6).

Another source of inputs to Monitor is COMMON, which contains all constants and parameters. COMMON is a block in high-order core storage of 175 locations. It begins at 77461_8 and proceeds downward in memory until the COMMON list is exhausted. It includes some dimension quantities. The composition of COMMON is described in Subsection 2.4, which also lists the FORTRAN routines that must be in core for successful operation of Monitor and its subprograms.

2.2 OUTPUT REQUIREMENTS

All outputs from the Monitor program are placed in COMMON and may be printed out on-line and/or read onto tape. The actual Monitor output data are used to write the two postflight reports specified by NASA-Space Task Group—the Quick Look and the Three-Day Report. Each report investigates four flight phases, Launch, Abort, Orbit and Re-entry, for stated periods of time. In addition, discrete events are reported.

At the beginning of each phase, for either report, a heading is printed containing the name of the phase, the Greenwich Mean Time of lift-off, and the longitude of Aries at lift-off, in degrees. The heading information is printed one time only for each applicable phase. The paragraphs following the headings contain the results of the postflight investigation in fixed-point format. (The headings and data formats are detailed in Appendix C of this manual.)

2.3 METHOD

This subsection deals with the manner in which Monitor organizes and controls the flow of information to produce the postflight reports.

NOTE: The major subprograms of Monitor are identified but not discussed in detail in this section. Details are given in the writeups of individual subprograms in Sections 3 through 6. Also, in the following discussion, the expression "main programs" is used synonymously with "Monitor."

At the start, Monitor calls the initialization program CHUMLY (see Subsection 3.1) to read into core four BCD output blocks. It then calls ACTORS (see Subsection 3.2) to place a series of constants into a common block called FACTOR. On the return from ACTORS, the main program prints a series of headings and data formats (see Appendix C) with on-line option. These headings and data formats actually contain the key for the postflight report.

Monitor then, from the on-line card reader, reads in three cards, which contain the physical and geographical constants* required on launch day, and calls INITIA (see Subsection 3.3). This initialization program converts these constants to forms more useful to Monitor. After returning from INITIA, the main program prints on-line, PLEASE CHECK INSTRUCTIONS TO DETERMINE SETTING OF SENSE SWITCH 6. SET AND PRESS START. The program comes to a PAUSE, with 77771_8 in the address portion of the storage register. After the operator presses START, the program tests sense switch 6 to determine whether the program shall proceed to SORTER (see Subsection 4.1). If the switch is up, a sort is done; if it is down, the sort is suppressed. Although the output of SORTER is necessary to the main program, it need not be redetermined if once obtained.

If a log tape is to be sorted, Monitor tells the operator to PLEASE CHECK TO SEE THAT NUMBER OF PHYSICAL LOG TAPES IS ENTERED IN ADDRESS PORTION OF KEYS. The program comes to PAUSE at 77774_8. After the operator presses START, the program transfers to SORTER.

On return from SORTER, the program determines whether the sort was successful. If the routine was unsuccessful, a message states that AN ERROR HAS BEEN DETECTED IN KEY OR TAPE SET-UP. PLEASE REVIEW OPERATING NOTES. AFTER COMPLETING CORRECTION PRESS START. The program, which is at PAUSE 77775_8 attempts to re-sort. If successful, the program is ready to continue. When no sort can be made, the program can go no further.

On the other hand, if sorted tapes are available, the operator is told to set them up and PLEASE CHECK TO SEE IF ALL TAPE UNITS ARE SET PROPERLY. IF SO, PLEASE PRESS START. There is a PAUSE 77772_8 until START is pressed, and Monitor calls the next subroutine, GETME (see Subsection 4.2).

The GETME subroutine searches a tape** created by SORTER for the occurrence of seven discrete events: GMTLO, SECO, times of Abort, Orbit and Re-entry initiate, number of retro-rockets fired and time first retrorocket fired referenced to lift-off.

Monitor returns from GETME and tests an indicator to determine whether lift-off has been found. If it has not been found, the main program assumes that

*The constants read in are EQURAD, FFLAT, GRAVIT, XMUE, CANJ2, CANJ3, CANJ4, AC, BC, RBAR, OMEG, AE, XLPAD, XLMI, HPAD, XLMO, XNUA, THETA0. (See Appendix A for FORTRAN symbols and definitions.)

**The tape created by SORTER is a high-speed input message tape. GETME searches 1,000 identical records in the second file of this tape.

there has been a set-up error. The operator is instructed to review his operating notes and correct the error. There is a PAUSE at 77773_8, and when the operator presses START, Monitor returns to GETME. If the GMT of lift-off (GMTLO) is not found, the main program cannot continue.

Using GMTLO, Monitor computes the Greenwich hour angle of Aries at lift-off and transfers to FIXIT1 (see Subsection 6.3) which ensures that the angle lies between 0 and 2π radians (including zero). The main program converts this to degrees and transfers to FIXIT (see Subsection 6.2), which ensures that this angle lies between zero and 360 (including zero).

Monitor tests whether the log tape contains the occurrence of six discrete events (all events searched for by GETME except the number of retro-rockets fired). If any of the six events have not been found, the program will continue with a zero value for each event. If the events have been found, HRSCNV (see Subsection 6.1) is called to convert these times from floating point seconds to fixed point integral hours, minutes, and seconds. FIXIT2 (see Subsection 6.4) ensures that the hours lie between 0 and 24 (including zero). The main program reports on-line either the time of occurrence of the individual event or the fact that it has not been found (i.e., not occurred).

After initialization has been completed, Monitor is ready to process the data and write its report. A request-for-report data card is read from the on-line card reader. This card indicates the type of report data that is requested. A zero in column one of the card indicates that the job is complete and that all data requested from the particular log tape are obtained. In this case Monitor writes on C3, the output tape, that the end of the job has been reached. An end of file is also written on C3, and the program comes to a PAUSE at 77776_8.

If the operator presses START, the program tests sense switch 2 to determine whether a log tape is to be processed. If sense switch 2 is down, the transfer is to the first statement and a log tape is used. If the switch is up, C3 is rewound and unloaded. The main program prints on-line that the JOB IS DONE and PAUSE's at 77777_8. If the operator now presses START the program transfers to the beginning to read and process a new log tape. Ordinarily, the operator would not press START, since there is usually only one log tape. The tapes would be dismounted for off-line printing.

The first character of the request-for-report card, if not a zero, should be a one or a two. One indicates a request for the Quick Look; two indicates a request for the Three-Day Report. The main program sets certain control values positive, converts the start and end times on the input card to seconds, and tests whether the time difference between the first and last times to be read is non-negative. If it is negative, the program prints on-line ERROR IN DECK SET-UP. TF EXCEEDS TL. REPUNCH, PRESS START. There is a PAUSE at 12121_8, and when the operator presses START, the program transfers back to read in the card again.

If the difference between the first and last times is positive, Monitor prints out a heading indicating the information that is included in the tapes created by SORTER. The main program also determines the number of observations to be taken from these tapes and the phase which is to be examined. The GMT of the first message is computed and the program prints on-line the GMT of lift-off and the longitude of the first point of Aries at lift-off. Monitor also computes the angle at 2-inch lift-off in the equatorial plane between Aries and the longitude of the GE radar, and the angle in the equatorial plane between Aries and the longitude of the pad.

The main program is about to enter the loop that starts writing the report (see connector #1 in Fig. 1-1). DONOUT (see Subsection 4.3) is called and high-speed output logged messages are read from the message tape (A4 or A5). The data are removed according to a predetermined time and flight phase.

There are four returns that can be indicated in a parameter JERROR of DONOUT's call statement. One indicates no more output on the log tape for the phase requested, the loop is terminated, and the next data card is read. Two indicates that high-speed output display quantities were output by the program but not transmitted through the DCC to Cape Canaveral. Three is the normal return; transmission has been successful. Indicator four shows a redundancy record. In this case, the main program skips this particular record and proceeds to the next time interval. Monitor saves the time produced by DONOUT (in general, this is the vector time from the log tape) to be incremented later in the loop.

When the indicator of DONOUT is either 2 or 3, Monitor, depending on the data source, either transfers to DONIN (see Subsection 4.4) or acts in the same manner as in the error return case of DONIN (see below). However, with processed data available, DONIN is used. DONIN processes the high-speed input message tape. In its call statement are the vector time, the data source, and an error indicator. When control is returned to Monitor, if the error indicator shows that DONIN cannot find the associated input message, the on-line printer states that THE PROGRAM HAS FAILED TO LOCATE INPUT MESSAGE CORRESPONDING TO CURRENT OUTPUT DATA. CURRENT TIME SINCE LIFT-OFF IS—.

If DONIN has not found the associated input message, Monitor sets to zero a number of parameters* which normally would be calculated by DONIN. Monitor then turns to core to utilize data deposited there by DONOUT. It determines which report is to be written—the Quick Look or detailed Three Day Report—and depending on the type of report required it selects and prints out the appropriate DONOUT data.

*These parameters, which are defined in Appendix A, are: X, Y, Z, X1, Y1, Z1, U, V, W, U1, V1, W1, VI, PSII, CL, VE, GE, PSIE, ASUBR, S, XMACH, QD, RN, and QS.

If DONIN has found the associated input message, subroutines IPCNV (see Subsection 5.2.1) or GECNV (see Subsection 5.2.2) are used to convert the coordinates of position and velocity in the appropriate data source reference frame (Impact Predictor 7090 Computer or Burroughs-General Electric Guidance Computer) to true inertial coordinates (see Appendix B). The LAUNCH (5.1.1) program is called, and it produces the calculated output required for the report.

When LAUNCH is entered and there is associated high-speed input, its duties may be divided among reporting launch, abort and re-entry displays. There can be associated high-speed input for an MA launch, an MR abort and an MA re-entry. If a re-entry switch is on, most of the LAUNCH program can be used for the MA re-entry situation. If an abort switch is on, RENTER (5.1.3), the re-entry processor must be used for an MA abort situation (there is no associated high-speed input).

After it has produced the required outputs, LAUNCH returns to the main program, which converts these outputs to their proper units for reporting. A heading is prepared and records are written. If the phase is Launch, the records are written according to the Launch Data Format in Appendix C. Detail, if required, is included in the report. If an abort situation is represented, the procedures for launch are identically applicable.

In general, the main program considers whether the output found by DONOUT was transmitted through the DCC. Referencing the indicator from DONOUT, Monitor utilizes sense lights 1 and 2 to determine whether there has been a change of status. If sense light 2 is on, all data have come from messages logged and not transmitted. If sense light 1 is on, all data have come from messages actually transmitted. The occurrence of one turns off the light of the other and this indicates a status change, which is reported on-line. If the status changes from a transmitting DCC to a non-transmitting DCC, the following message appears on-line: STATUS CHANGE—REPORT DERIVED FROM NON-TRANSMITTING DCC. CURRENT ELAPSED TIME SINCE LIFT-OFF TIME IS—. In the contrary situation, the message without the prefix NON is presented. Time is incremented and the main program returns to DONOUT to pick up the next message. (This completes one pass through the loop.)

If the phase is Orbit, the main program calls ORBIT (see Subsection 5.1.2). This subroutine produces a number of parameters needed by the postflight report for the Orbit phase. Orbit is characterized by the lack of high-speed input data. Therefore, DONIN is not used by Monitor, but after the main program receives the return from DONOUT, ORBIT is called.

With the return to Monitor from ORBIT, certain parameters in the report are converted to appropriate units, as is done with LAUNCH. The report is written. If detail is required, the additional data available from DONOUT are converted and written. An examination is made of the status with respect to transmission or non-transmission through the DCC. The appropriate message is printed indicating the change of status. Time is incremented and the loop is closed.

If the phase is re-entry, upon return from DONOUT Monitor calls RENTER (see Subsection 5.1.3) and indicates whether input data are available. If input data, are available, the re-entry switch is put ON and LAUNCH is called. If input data are not available, this indicates a true re-entry situation and RENTER proceeds to produce the appropriate parametric values.

After this phase is processed, a new card is read until a data card with a zero in column one is read. As mentioned earlier this is the end card. The message, THE END, is written; an End-of-File is placed on the output tape, C3, and the program comes to a PAUSE.

There is now an option. If sense switch 2 is down, processing of another log tape may begin without the reloading of the program into core storage. If, however, sense switch 2 is up, the main program rewinds and unloads the C3 tape after START is pressed. The message, THE JOB IS DONE, is printed, and a PAUSE occurs at 77777_8. At this stop, the job may be pulled off the machine or START may be pressed and control returned to ACTORS for a completely new iteration. (There is no final STOP in the Monitor program.)

2.4 USAGE

a) Storage

The Postflight Monitor and its subprograms use shared storage locations, COMMON, for all input and output. The contents of COMMON are arranged in the following order:

X, Y, Z, X1, Y1, Z1, U, V, W, U1, V1, W1, XIP, YIP, ZIP, XIP1, ZIP1, YI1, XX1, YY1, ZZ1, XI, ETA, ZETA, XI1, ETA1, ZETA1, VI, VE, GI, GE, PSII, PSIE, RBAR, RRBAR, RADIUS, ARBAR, AXIS, HE, HA, DTHTAP, R01, XLRHO, GMTLO, BCAC, RPBAR, EQURAD, FFLAT, GRAVIT, XMUE, CANJ2, CANJ3, CANJ4, AC, OMEGAE, XLPAD, XLM1, HPAD, XLMO, XNUA, THETA0, DL, CL, XL, RL, ASUBR, XMACH, QD, RN, S, D, CR, SR, QS, VIVR, VIVRGE, VIVRIP, GIGE, GIIP, ECC, XINC, ARGP, ARGP1, OME, OME1, TP, TA, EA, XMA, PER, PHIMIN, XLMMIN, PHIMAX, XLMMAX, PHIIP, XLMIP, DLMI, CS, DENS, XNU, P, NUMORB, NORBCP, XLAMP, DPHIR, JAREA, NOGOGO, DTR, EGT, ECTRS, GTRS, GTL, XICTRC, GMTLC, GMTRC, GMTRC1, GMTRC2, GMTRC3, ECTRC, ECTRC1, ECTRC2, ECTRC3, GMTRS, T, TSECO1, NORET, TRETRO, KHR, KMIN, KSEC, FACTOR, A, B.

Included in the listing of COMMON are six dimension quantities which involve more than one location. They appear below with their respective dimensions.

KHR - 10
KMIN - 10

KSEC	-	10
FACTOR	-	25
A	-	8
B	-	8

b) FORTRAN Programs

The Postflight Reporter is executed with the FORTRAN 32K Monitor. In addition to the FORTRAN Monitor, the following FORTRAN programs must be read into core for the successful operation of the main program and its subprograms:

(FPT), (CSH), (SLI), (RTN), (STH), (SLO), (FIL), (SPH), (EFT), EXP(2, EXP, SQRT, EXP(3, ATAN, SIN, COS.

For column binary operation, the FORTRAN Monitor automatically supplies these. For row-binary (absolute) operation, these must be a part of the operational program.

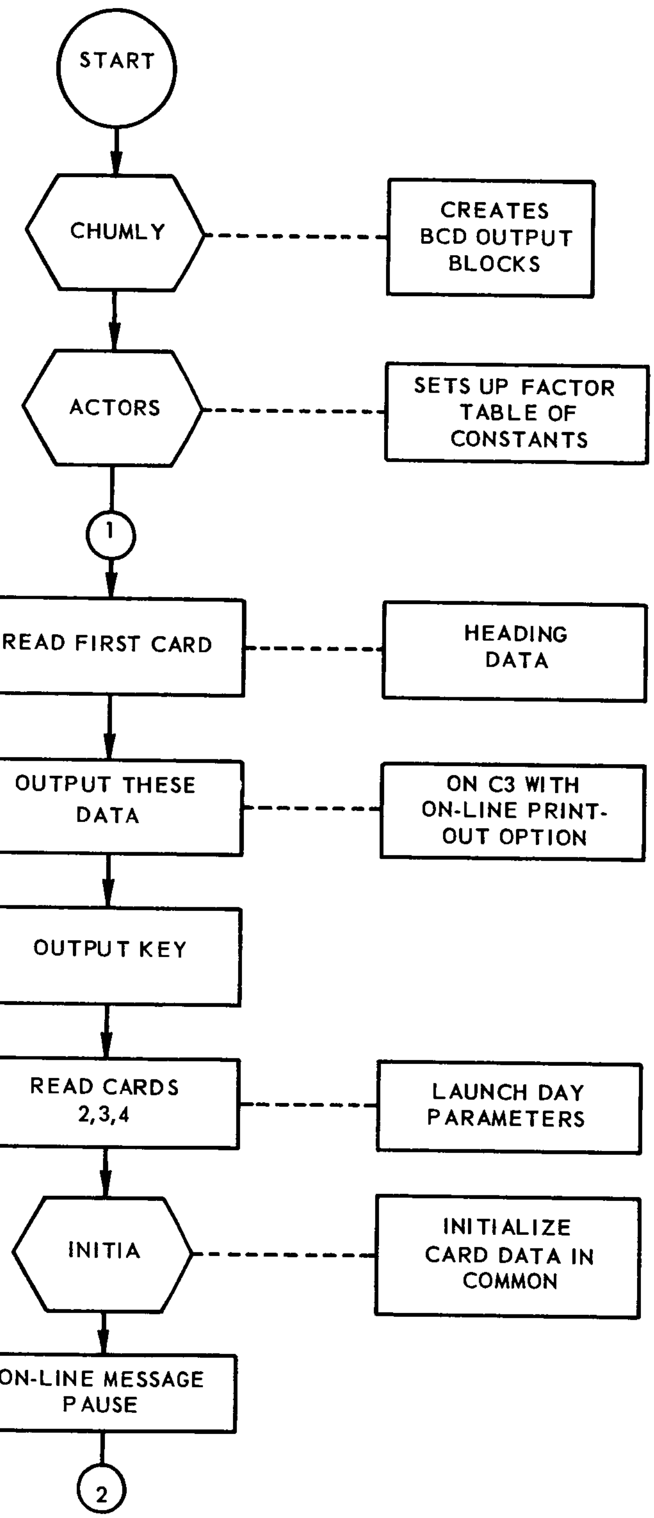


FIGURE 2-1. POSTFLIGHT MONITOR FLOW CHART *(Sheet 1 of 3)*

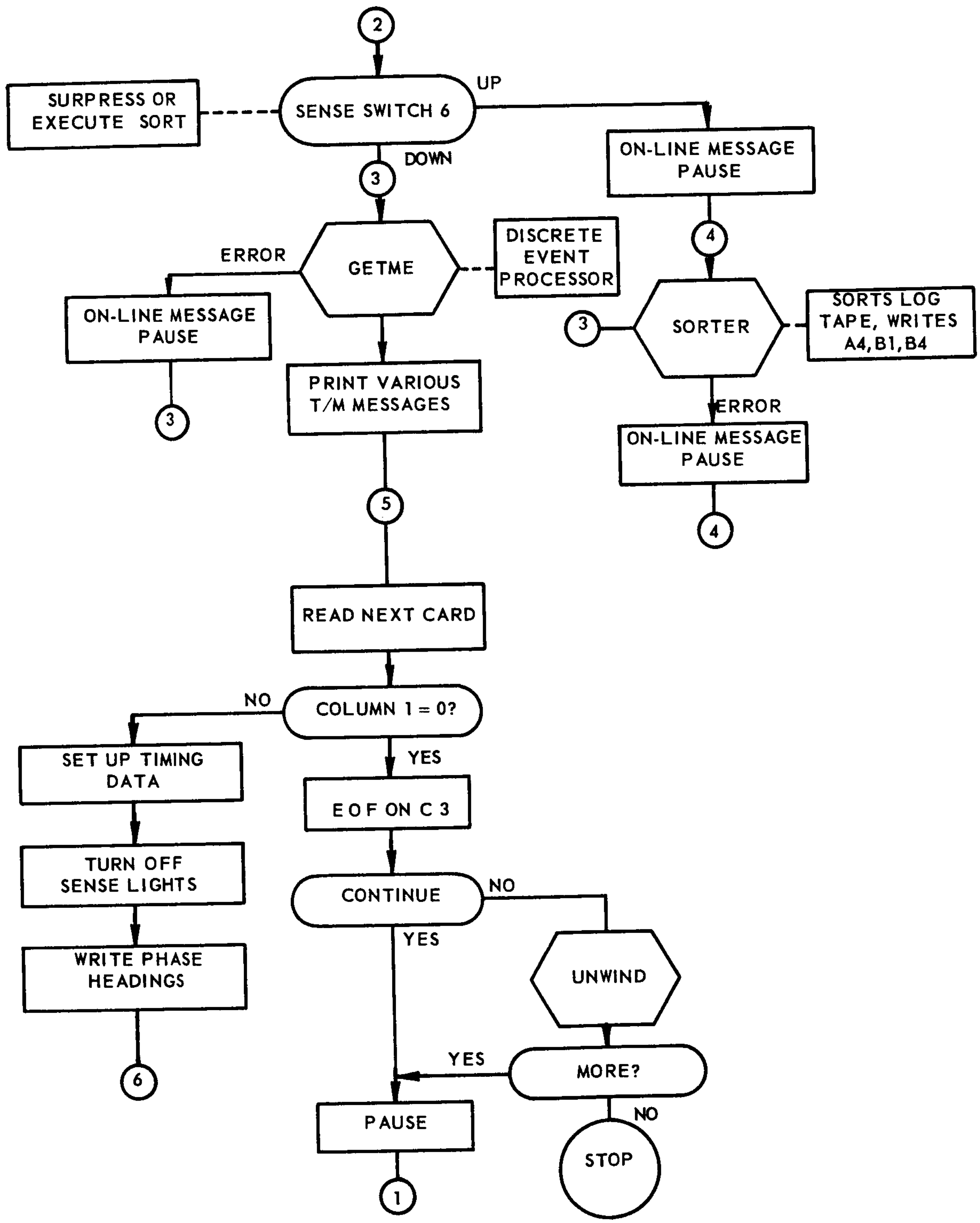


FIGURE 2-1. POSTFLIGHT MONITOR FLOW CHART *(Sheet 2 of 3)*

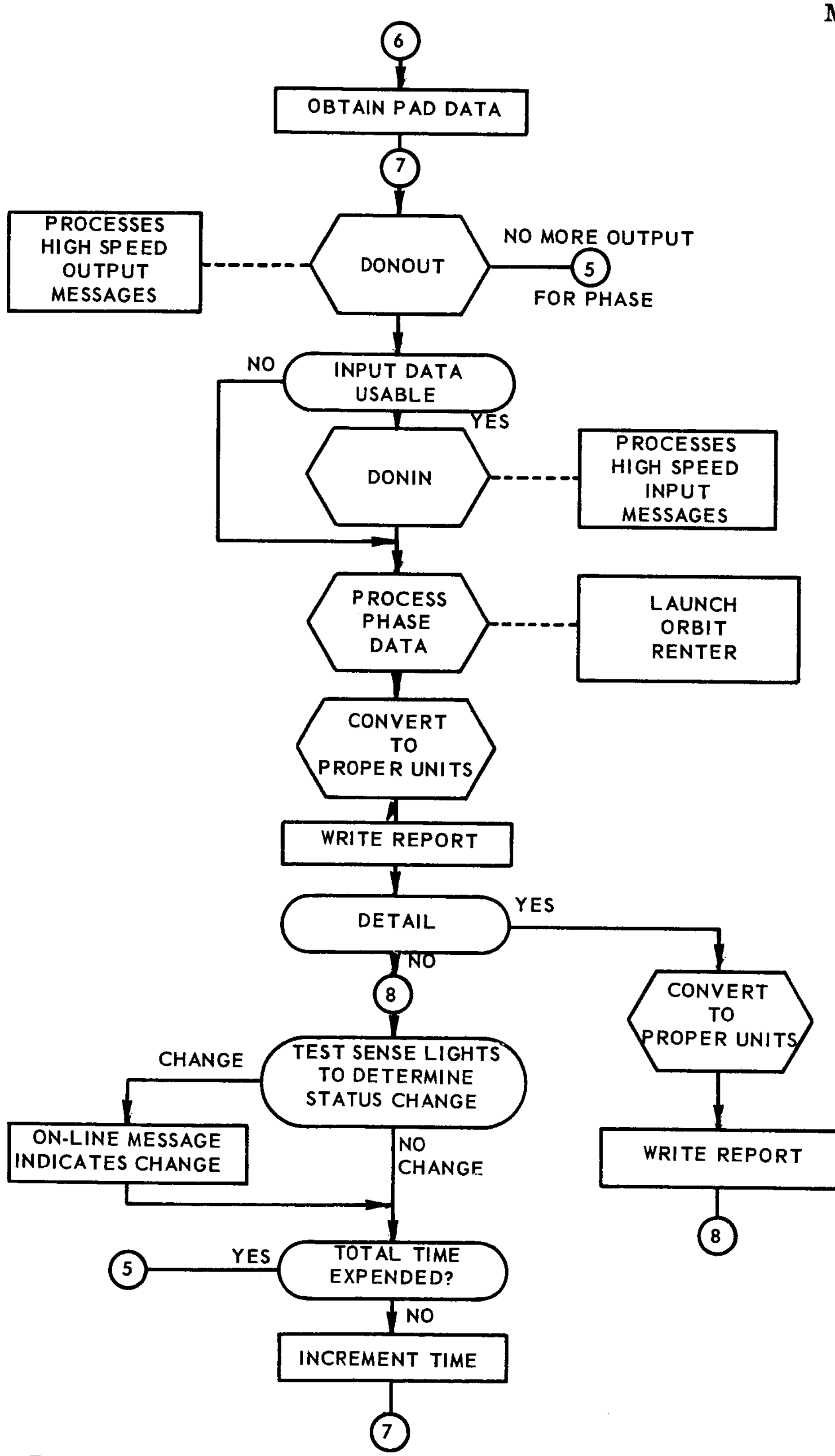


FIGURE 2-1. POSTFLIGHT MONITOR FLOW CHART *(Sheet 3 of 3)*

SECTION 3
INITIALIZATION PROGRAMS

There are three initialization programs. They establish, at the beginning of the Postflight Reporter Program, certain data and values used by subsequent processor programs. These programs—CHUMLY, ACTORS, INITIA—are described below in Subsections 3.1 to 3.3. Their relationship to the overall Postflight Reporter Program is shown in the flow chart Fig. 2-1.

3.1 BCD OUTPUT INITIALIZATION PROGRAM (CHUMLY)

CHUMLY creates four BCD output blocks for the Monitor program. The contents of these blocks are read into core storage, providing the BCD information needed for the report. They are:

HED1	Phase Title (Launch, Abort, Orbit, Re-entry)
HED2	Transmission Title (DCC Transmission Status)
HED3	Data Source Title (IP 7090, B-GE, Raw Radar)
HED4	GO, NO-GO Title (GO, NO-GO, GO and NO-GO, NOT COMPUTED)

3.2 CONSTANT FACTORS INITIALIZATION PROGRAM (ACTORS)

ACTORS reads into core the constant values of certain conversion factors used throughout the Postflight Reporter Program. These constants read into a common block, FACTOR, are:

Feet per statute mile (5280)

Feet per nautical mile (6076.1155)

Seconds per Mercury unit of time (806.8104)

Kilograms per meter^3/slugs per foot^3(515.378725)

Meters per foot (0.3048)

Two pi (6.2831853072)

One radian (57.2957795 degrees)

ACTORS also initializes the following FACTOR values to minus zero (-0):

Time since lift-off (in floating-point seconds) of start of orbit phase

Time since lift-off (in floating-point seconds) of start of abort phase

Time since lift-off (in floating-point seconds) of start of re-entry phase

Geocentric latitude of capsule at end of retrofire

Longitude of capsule at end of retrofire

Local earth radius of capsule at end of retrofire

3.3 SYSTEM PARAMETER INITIALIZATION PROGRAM (INITIA)

INITIA converts system parameters required on launch day into the basic internal units needed for use in subsequent programs of the Postflight Reporter. The parameter values obtained by INITIA are:

Geodetic latitude of launch pad in radians

Geocentric latitude of launch pad in radians

Geocentric radius at pad in feet

Total distance from geocenter to pad in feet

Longitude of launch pad in radians

Longitude of GE radar in radians

Launch azimuth in radians

Greenwich hour angle at midnight preceding launch in radians

Zeroth, second, third and fourth harmonics of earth's potential in English units

INITIA also sets up eight coefficients of each seventh degree polynomial fit (high and low altitude fits) of atmospheric density. The coefficients, determined by a least squares criterion, are placed in a common block for use by the program ATMOS (see Subsection 5.2.6).

SECTION 4
TAPE PROCESSOR PROGRAMS

All information for the Postflight Reporter Program is derived from the logical log tape created during operation of the Mercury Program System. The programs discussed in this section are concerned with the processing of the Mercury log tape and the tapes created from logged data. These programs set up the information required by the phase processor programs for preparing the postflight reports. The SORTER program consists of a control program and nine subroutines. SORTER processes the log tape and creates at least three other tapes from logged information.

The other programs process the tapes created by SORTER. GETME processes the discrete events on the second file of the high-speed input tape created by SORTER. DONIN and DONOUT process the SORTER high-speed input and output tapes according to predetermined times and flight phases.

4.1 LOG TAPE SORT PROGRAM (SORTER)

SORTER performs the unpacking and processing of the logical tape. The program, using the Mercury Program System log tape(s), creates at least three other tapes as outputs. Nine subprograms provide input to SORTER. The flow chart for SORTER is shown in Figure 4-1.

4.1.1 SORTER Control

The operation of the SORTER program, including the calling of the subroutines, is controlled by the following procedure.

4.1.1.1 Input Requirements

Input to SORTER is the log tape on B6 and the one on B7, if necessary. The other input is the key setting giving the number of physical log tapes which comprise the logical log tape. The following subroutines are used with this program: GEB, IPORR, MANIN, HSOP1, BCTB (BCTB1), BCTBI (BCTBJ), TISWS, HMSTS, and CNV1 (GCNVE) (see Subsections 4.1.2 through 4.1.10, respectively).

4.1.1.2 Output Requirements

This program creates at least three tapes from the Mercury Program System logical log tape: a miscellaneous tape (B1), a high-speed input tape (B4), and a high-speed output tape (A4). If necessary, there may be a second high-speed output tape (A5). All redundant and rejected records are placed on the B1 tape for future analysis.

4.1.1.3 Method

SORTER first rewinds the A4, B1 and B4 tapes before reading data onto them. The entry keys are stored and sensed to determine if they contained the the number of physical log tapes comprising the logical log tape. If the keys were not set. the program stores an error indication and returns to the main Post-flight Monitor program.

If the keys were set, SORTER initializes to minus one the temporary storage locations for discrete events. All indicators and switches are also set to zero. The program then reads a record from the log tape and tests for a redundancy. If a redundancy occurs, the program attempts to read the record ten times. If the redundancy continues to occur, the record is written on the B1 tape and the next record is read.

If the redundancy does not continue or if there was none, the record is tested for an end of file. If the record is not an end of file, the log identification for the block is stored in an identification buffer. SORTER tests the subchannel number associated with the block. If the subchannel number equals one (1), the program calls the subroutine GEB (see Subsection 4.1.2). IPORR (see Subsection 4.1.3) is called if the subchannel number is two (2). HSOP1 (see Subsection 4.1.5) is called if the subchannel number is three (3). If the subchannel number is 30, the program calls MANIN (see Subsection 4.1.4.6).

The Mercury Program System log tape contains ten 17-word blocks of data per record. If SORTER finds that the subchannel number is not 1, 2, 3 or 30, the program determines whether all ten blocks of data have been processed. If the blocks have not been processed, the program continues to the next block of data. If all ten blocks have been processed, the program proceeds to the next record.

If the record being tested is an end of file record, it is determined whether this is the logical end of the log tape. If this is not the logical end of the tape, the program examines to see whether the B6 or B7 was processed. The tape that was processed is rewound and unloaded. All references are changed to the other tape.

If the end of the logical log tape is reached, SORTER determines whether B6 or B7 was the last tape processed. This last tape is rewound and unloaded. If all tapes have been processed, success is indicated and a search is made for seven discrete events for the program GETME (see Subsection 4.2). All data pertaining to these events are written 1,000 times on the second file of the B4 high-speed input tape.

SORTER finally determines whether the last output record was written on A4 or A5 and writes a double end of file on the appropriate tape. An end of file is also written on B1. Then, the tapes are rewound and B1 and B6 are unloaded.

4.1.1.4 Usage

Call Statement

CALL SORTER (NOYES)

NOYES is an error indicator

1 indicates error return, error in tape setup

2 indicates normal return

4.1.2 Subchannel 1 Processing Program (GEB)

GEB processes all subchannel 1 data (high-speed B-GE input from Cape Canaveral). From the output telemetry of this Canaveral data, it determines the occurrence of discrete events. This program also converts position ($\bar{r}$) and velocity ($\vec{v}$) vectors to internal units of the Postflight Reporter. The flow chart for GEB is shown in Figure 4-2.

Using the subroutine IOHSGB, (see Volume MC 105, Goddard Processor Programs) GEB receives a 170-word record to be processed and outputs data on the B1 (miscellaneous) and B4 (high-speed input) tapes. Each message consists of four 17-word blocks, and GEB determines which of these is to be processed. If it is the first block of the sequence, the 12 significant data words are stored and the program returns to process the next block. If it is the second third, or fourth 17-word block, a log time test is made.

The log time of each block is compared with that of the first block. If the times are equal, each of 12 significant data words are stored sequentially until a 48-word buffer is filled. The log time is stored in a location used by IOHSGB, and GEB calls IOHSGB.

If the log times are not equal, all "ones" (1's) are stored in the remainder of the output buffer. The output buffer is then written on the B1 tape. The log identification is stored in the output buffer and the log time is placed in a location used by IOHSGB.

If IOHSGB finds an error in the computed data or in the telemetry, it exits to location MFML6A. If IOHSGB finds computed data without error, it exits to MFHSGB.

When exit is to MFHSGB, the vector time is stored. The r and v values are converted to feet and feet per second, respectively, and are also stored. If lift-off has occurred, GMT of lift-off is stored and the lift-off indicator is set to nonzero. If lift-off has not occurred, the program processes the next block of data.

If SECO has occurred and if the SECO indication did appear, the time of SECO is stored and the SECO indicator is set to non-zero.

After lift-off is found, the output is written on B4 and it is determined whether this is the end of the tape. If it is the end of B4, an end of file is written and B4 is rewound and unloaded. Whether or not an end of tape appears, GEB processes the next block of data.

When exit is to MFML6A, it is determined whether an error occurred in the data and, if so, whether the error was due to the previously mentioned time test. If the error was a result of the time test, counters, indicators, and addresses are reset and GEB restarts processing. If the error did not result from the time test, the data is read into an error buffer, error is indicated and written on the B1 tape, and GEB returns to SORTER.

If an error did not occur, the data is presumed to be telemetry data, and the time tags, discrete signals, B-GE selected source, elapsed capsule time and retrofire clock setting are stored. GEB then tests for lift-off, abort initiate, SECO, retro-rocket firings, abort inception, and orbit inception.

If lift-off occurred in the message being examined, then GMT of lift-off is stored and the lift-off indicator is set. If neither abort initiate nor SECO has occurred previously but occurs in the present message, GMT of abort initiate is stored and the SECO and abort initiate indicators are set.

The program then tests whether the first, second, or third of three retro-rockets have already been fired. If the first or second or none of the retro-rockets have been previously fired, GEB tests the present message for an indication of retro-rocket firing. If the retro-rockets did fire, the time of retro-fire and the number of retro-rockets already fired are stored.

If the retro-rockets did not fire or if the firing of the third retro-rocket was previously indicated, the program tests for abort and orbit phase inception. If either occurred previously or if either cannot be found in a previous or in the present message, GEB returns to continue processing. If either is found in the message under examination, the indicators are set to non-zero, the GMT's of inception of the events are stored, and the program returns.

4.1.3 IP 7090 High-Speed Input Processor Program (IPORR)

IPORR processes all high-speed input data from the Cape Canaveral IP 7090 Computer. This program receives 17-word blocks of data from an input record. Using the subroutine IOHS09 (see Volume MC 105, Goddard Processor Programs), IPORR process the IP 7090 high-speed input and places significant data on the high-speed input tape (B4). The erroneous data is packed and stored in an error buffer and written on the miscellaneous tape (B1). The flow chart for IPORR is shown in Figure 4-3.

When IPORR is entered and a block of data is input, the identification of the block is stored in an identification buffer and the program determines whether this is the first, second, third or fourth block of a sequence. If it is the first block, the 12 significant data words are stored in the I0HS09 input buffer. If it is the second, third or fourth block, a time test is made.

The time associated with each sequential block is compared with the time of the first block. If the times agree, the data in the block being compared with the first block is stored sequentially until a 48-word buffer is filled. If the times do not agree, all "ones" (1's) are stored in the remainder of the buffer and written on the B1 tape. An error indicator is also set. Whether or not the times are equal, the identification of the block is stored in an output buffer and the log time is stored in a location used by I0HS09, which is then called.

I0HS09 exits to MFHS09, if the data source is either raw radar or IP 7090 processed data. If the data source is raw radar, the program returns to process the next block. If the data source is the IP 7090, the vectors are converted and stored with respective time tags in the output buffer. The program tests for lift-off at this time. If lift-off did not already occur, the program returns to process the next block. If lift-off did occur, the output buffer is written on the B4 tape and the program returns to process the next block.

I0HS09 exits to MFHS08, if the data is assumed to be either telemetry or erroneous. If the data is erroneous and the error is a result of the previously mentioned time test, the program returns. Ir the error is not from the time test, the data is packed, stored, and written on B1. The program returns to SORTER.

If no error is indicated, time tags, discrete messages, IP selected source indicator, elapsed capsile time, and retro-fire clock setting are stored. The program tests for lift-off. If it has not occurred previously, but does occur in the message being examined, GMTLO is stored and the lift-off indicator is set to non-zero.

IPORR also tests for SECO, abort initiate, first, second and third retro-fire, abort phase inception, and orbit phase inception. If each of these is found, the elapsed time since lift-off of GMT of its occurrence is stored and the appropriate indicator is set to non-zero. Whether or not the discrete events occurred, the program returns to SORTER to process the next block of data.

4.1.4 Manual Insertion Processor PProgram (MANIN)

MANIN searches manually inserted messages for the occurrence of discrete events. This program uses I0MANI (see Volume MC 105, Goddard Processor Programs) as a subroutine. The flow chart for MANIN is shown in Figure 4-4.

The program is entered and a word count is taken from the log tape and stored in the decrement of a location already containing the address of the first word of an input buffar. The data is stored and the location is placed into the AC. MANIN call I0MANI to process the input data.

If lift-off data were in the message, I0MANI exists to MFMAN1. Thereupon MANIN converts GMT of lift-off (GMTLO) to floating point and stores it. A one (1) is stored in the lift-off indicator. This GMTLO overrides any previous GMTLO found by any other subroutine. The program returns to process the next block of data.

An exit to MFMAN2 indicates retro-fire data. The number of retro-rockets fired is stored. A three (3) is placed in the retro-fire indicator, and the GMT of the first retro-fire is converted to floating point and stored. This overrides any previous retro-fire time.

An exit to MFMAOS indicates orbit or abort data, or both. If abort did not already occur but if abort inception did occur in this message, a one (1) is stored in the abort inception indicator. The log time associated with the message is converted to floating point and stored. The program returns to SORTER to process the next block of data.

If abort inception occurred previously, it is determined whether the orbit or abort switch was thrown. In either case, MANIN returns to process the next block of data until all the blocks given as input are processed.

4.1.5 High-Speed Output Tape Writer Program (HSOP1)

This program processes all high-speed output data and writes it on the A4 output tape. HSOP1 converts and scales the data with five conversion subroutines (see Subsections 4.1.6 through 4.1.10). The flow chart for HSOP1 is shown in Figure 4-5.

Input to this program are two 17-word data blocks. HSOP1 also uses the conversion subroutines BCTB (BCTB1), BCTBI (BCTBJ), HMSTS, TISWS, and CNV1 (GCNVE). The program outputs a 44-word buffer and an A4 tape.

The program determines whether lift-off has occurred. If it did not occur, the program returns to process the next block of data. If it did occur, the log identification, sub-frame indicator, and phase indicator are stored. If the first frame is being examined, the significant data are stored in the first 12-words of a temporary storage buffer. The odd-even indicator is masked and stored, and the program returns to process the next block.

If the second frame is being read, significant data are stored in the second 12 words of the temporary storage buffer. If the message is an odd frame, strip chart data, data source indicator, and wall map data are all masked and stored in the output buffer. Digital displays are masked. GTRS and GMTLC are converted by HMSTS; ϕ_{IP} is converted by BCD; λ_{IP} is converted by BCTB1; $r - \overline{R}$, i, and V/VR are converted by BCTBI; and γ is converted by BCTBJ. All converted values are stored in the output buffer, and all data from the plotboards are also stored there. CNV1 (GCNVE) then converts all non-digital display data to standard values (radians, feet, seconds). All data in the output buffer are converted to floating-point data, according to phase. The floating-point output buffer data are written on A4.

If the message is an even frame, data are masked and stored, as for the odd frame message. Digital displays are masked and converted. $GMTRC_s$, $ECTRC_s$, Δt_r, and GMTRS are converted by HMSTS; ICTRC is converted by TISWS. All converted values are stored in the output buffer, and all data from the plotboards are also stored there. CNV1 (GCNVE) converts all non-digital display data to standard values. All data in the output buffer are converted to floating point data, according to phase, and these output buffer data are then written on A4.

4.1.6 <u>BCD Word Conversion Program (A)</u> (BCTB/BCTB1)

This subroutine of HSOP1 converts modified BCD numbers to fixed point binary numbers. The modified BCD numbers are in one of two prescribed formats, which are prerequisites for proper entry into the subroutine. The flow chart for BCTB/BCTB1 is shown in Figure 4-6.

This program is entered through BCTB if the format of the modified BCD number is 4 bits/digit, 4 bits/digit, 3 bits/digit and 4 bits/digit. Entry is through **BCTB1 if the format of the** word to be converted is 1 bit/digit, 4 bits/digit, 4 bits/digit, 3 bits/digit and 4 bits/digit.

The modified BCD number should be in the AC at time of entry. The converted binary digits are summed and placed into the AC upon exit.

The calling sequences for both entries are:

TSX $BCTB (or $BCTB1)

$\alpha + 1$ Normal Return

4.1.7 <u>BCD Word Conversion Program (B)</u> (BCTBI/BCTBJ)

This subroutine of HSOP1 converts modified BCD numbers to fixed point binary numbers. The modified BCD numbers are in one of two prescribed formats, which are prerequisites for proper entry into the subroutine. The flow chart for BCTBI/BCTBJ is shown in Figure 4-7.

The modified BCD numbers are in 4 bits/digit format. If the input number is not preceded by a sign, the subroutine is entered through BCTBI. If the number is preceded by a sign, the subroutine is entered through BCTBJ. In this latter case, the appropriate sign is set in the AC. However, both subroutines convert, sum the digits and place the fixed point binary numbers into the AC upon exit.

The calling sequences for both entries are:

TSX $BCTBI (or $BCTBJ)

$\alpha +1$ Normal Return

4.1.8 Time Word Conversion Program (A) (TISWS)

This subroutine of HSOP1 converts a modified BCD number representing a time (in hours, minutes and seconds) to a binary number representing the same time (in fixed point seconds). The flow chart for TISWS is shown in Figure 4-8.

The modified BCD number, placed in the AC when the subroutine is entered, is given in the following format:

1 bit	-	sign for hours
2 bits/digit	-	tens of hours
4 bits/digit	-	unit of hours
1 bit	-	sign for minutes
3 bits/digit	-	tens of minutes
4 bits/digit	-	unit minutes
1 bit	-	sign for seconds
3 bits/digit	-	tens of seconds
4 bits/digit	-	unit seconds

Each digit is converted and/or scaled to its appropriate equivalent in fixed point seconds with the correct sign. The binary equivalents (with sign) are summed and stored in the AC when the program exits.

The calling sequence is:

TSX $TISWS

$\alpha +1$ Normal Return

4.1.9 Time Word Conversion Program (B) (HMSTS)

This subroutine of HSOP1 converts a modified BCD number representing a time, to a binary number representing the same time (in fixed point seconds). The flow chart for HMSTS is shown in Figure 4-9.

The modified BCD number, placed in the AC when the subroutine is entered, is shown in one of the following two formats:

Format 1	Format 2
2 bits/digit - tens of hours	2 bits/digit - tens of hours
4 bits/digit - unit hours	4 bits/digit - unit hours
3 bits/digit - tens of minutes	3 bits/digit - tens of minutes
4 bits/digit - unit minutes	4 bits/digit - unit minutes
	3 bits/digit - tens of seconds
	4 bits/digit - unit seconds

Each digit is converted and/or scaled to its appropriate equivalent in fixed point seconds. The binary equivalents are summed and stored in the AC when the program exits.

The calling sequence is:

```
     TSX    $HMSTS
α +1 Normal Return
```

4.1.10 Units Conversion Program (CNV1/GCNVE)

This program is used by HSOP1 to convert and scale high-speed output from granular values to significant standard values. The high-speed output may be from the strip chart (launch phase only), plotboard 3 (orbit phase only), or from the wall map and plotboards 1, 2 and 4 (all phases). The flow chart for CNV1/GCNVE is shown in Figure 4-10.

The address of the first location in the buffer to be converted is given in the PZE of the calling sequence. For launch phase, there is a specific location. All other phases are combined in another separate location. The buffer contains all high-speed output data for conversion when the program is entered either through CNV1 (for launch phase) or GCNVE (for all other phases). The AC contains a phase indicator and the MQ contains an odd-even indicator.

The buffer area contains all converted and scaled high-speed output data when the program returns control to HSOP1.

The calling sequence is:

```
     TSX    $CNV1 (or $GCNVE)
     PZE    (address of first location in buffer to be converted)
α +2 Normal Return
```

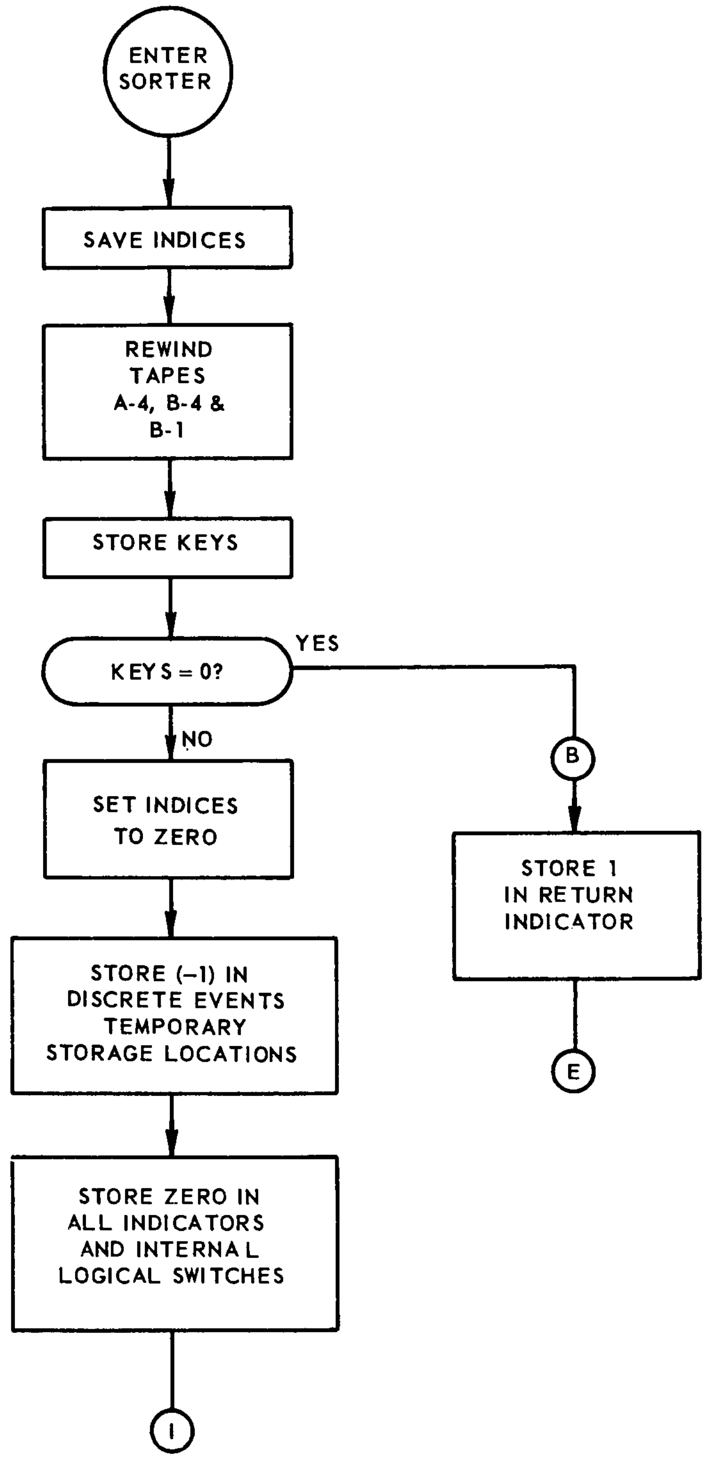


FIGURE 4-1. LOG TAPE SORT PROGRAM (SORTER CONTROL) *(Sheet 1 of 4)*

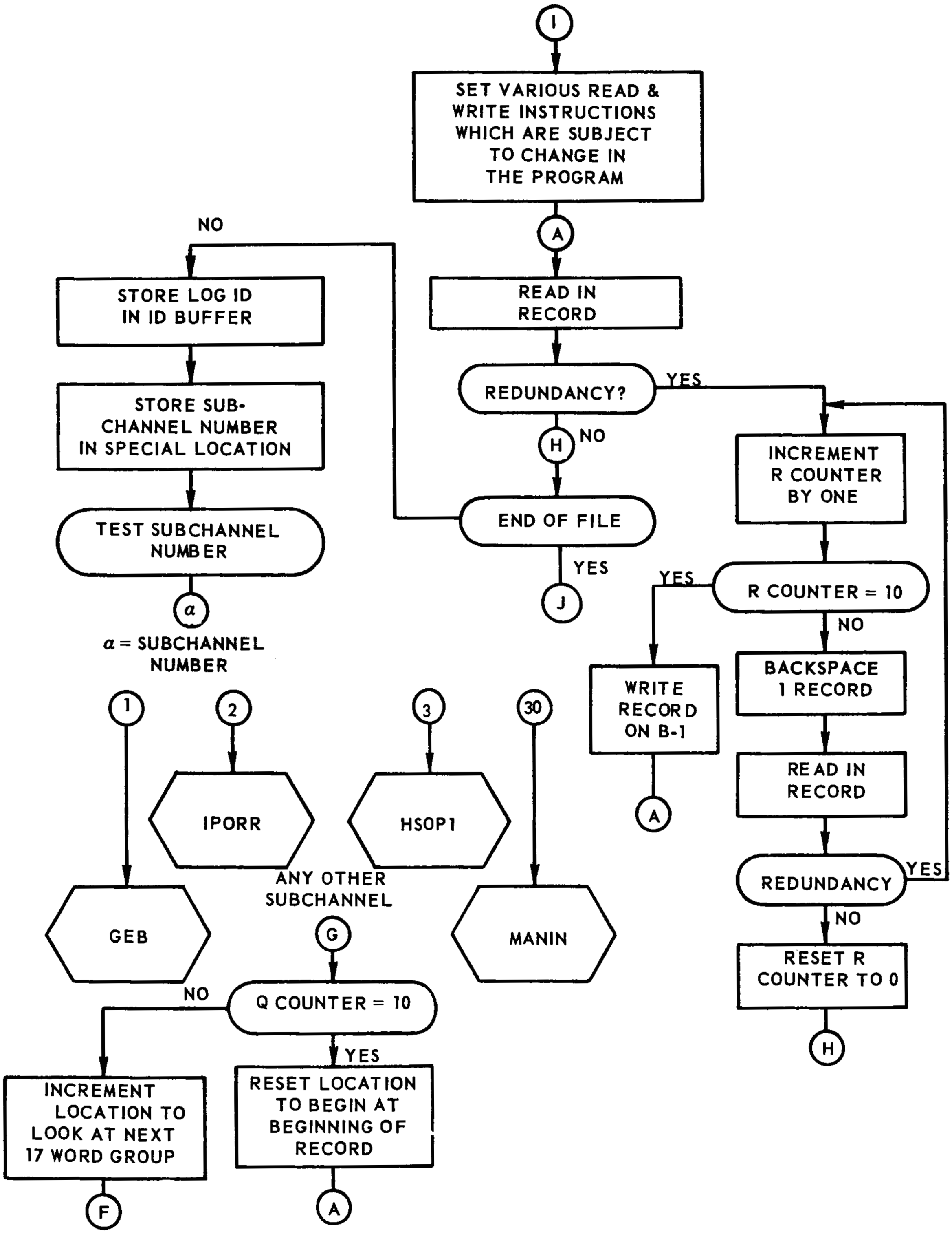


FIGURE 4-1. LOG TAPE SORT PROGRAM (SORTER CONTROL) *(Sheet 2 of 4)*

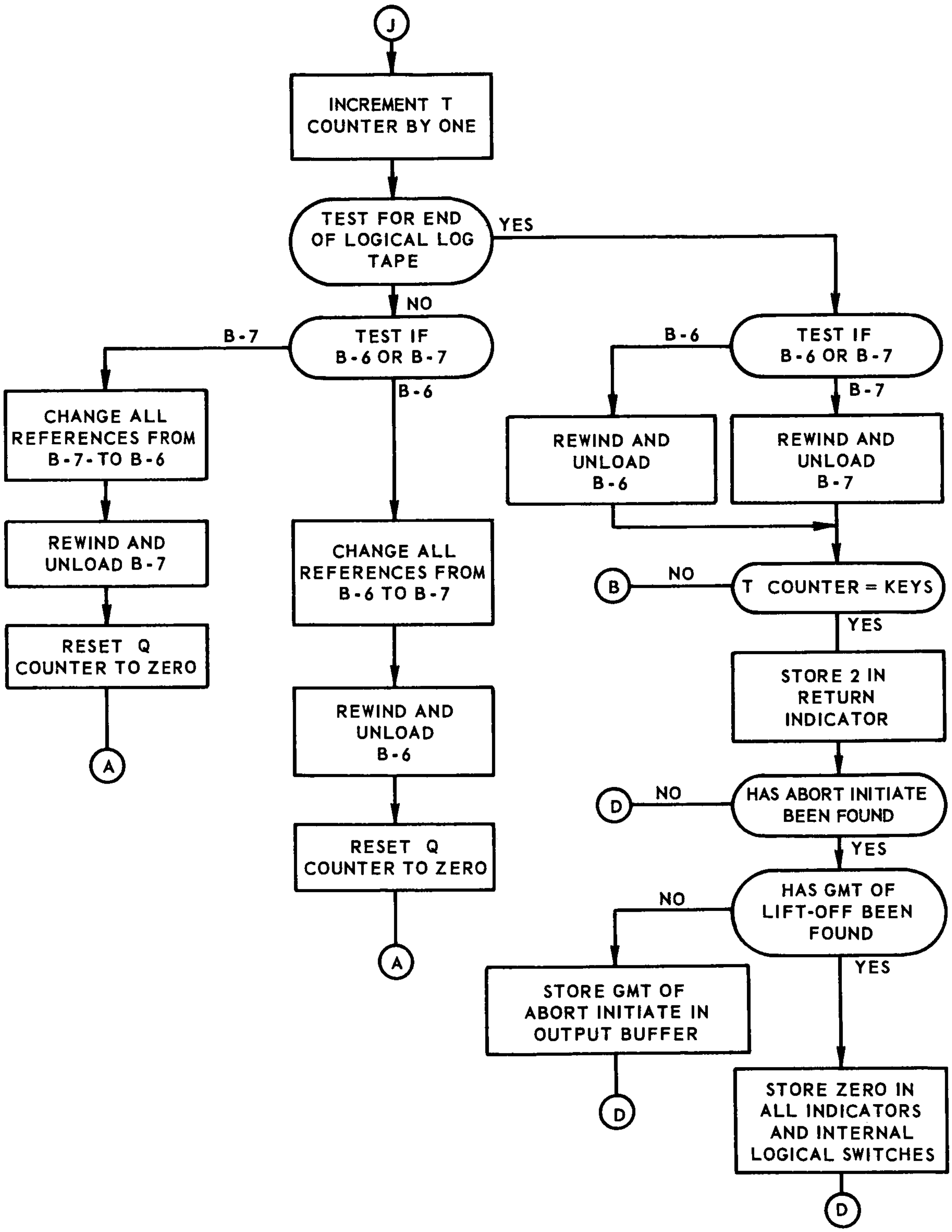


FIGURE 4-1. LOG TAPE SORT PROGRAM ISORTER CONTROL *(Sheet 3 of 4)*

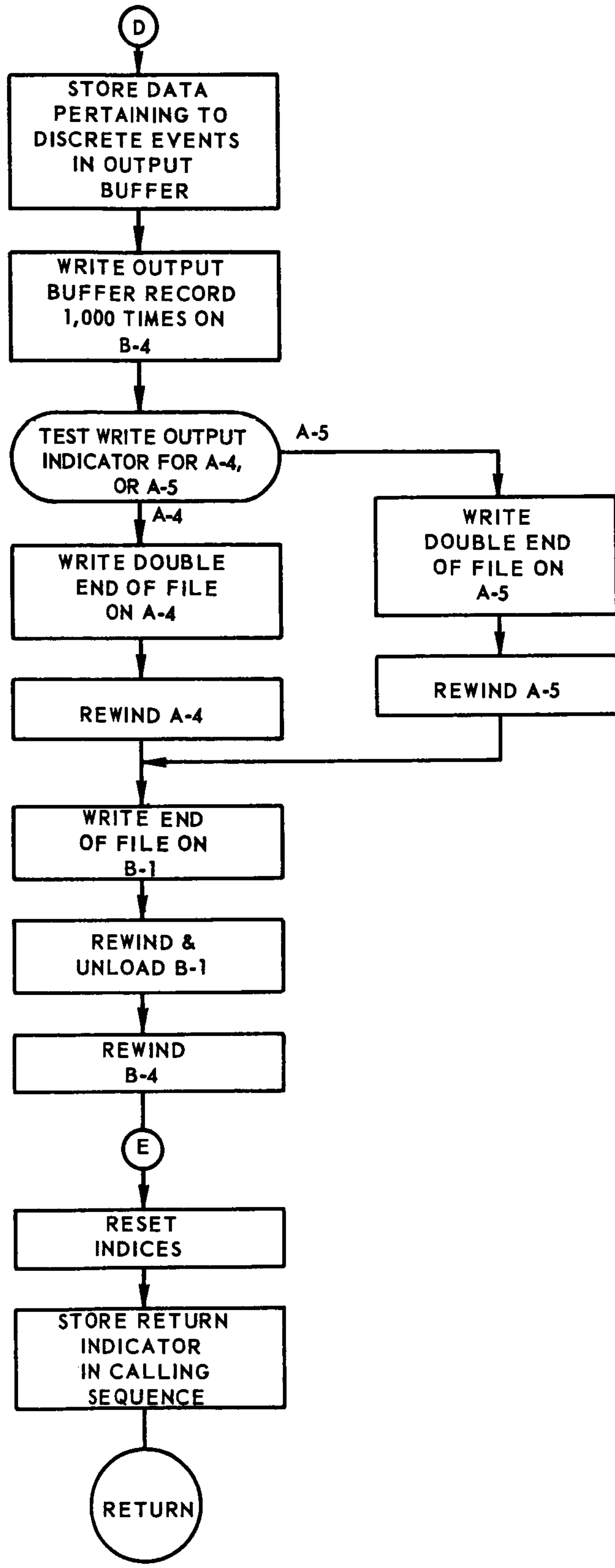


FIGURE 4-1. LOG TAPE SORT PROGRAM (SORTER CONTROL) *(Sheet 4 of 4)*

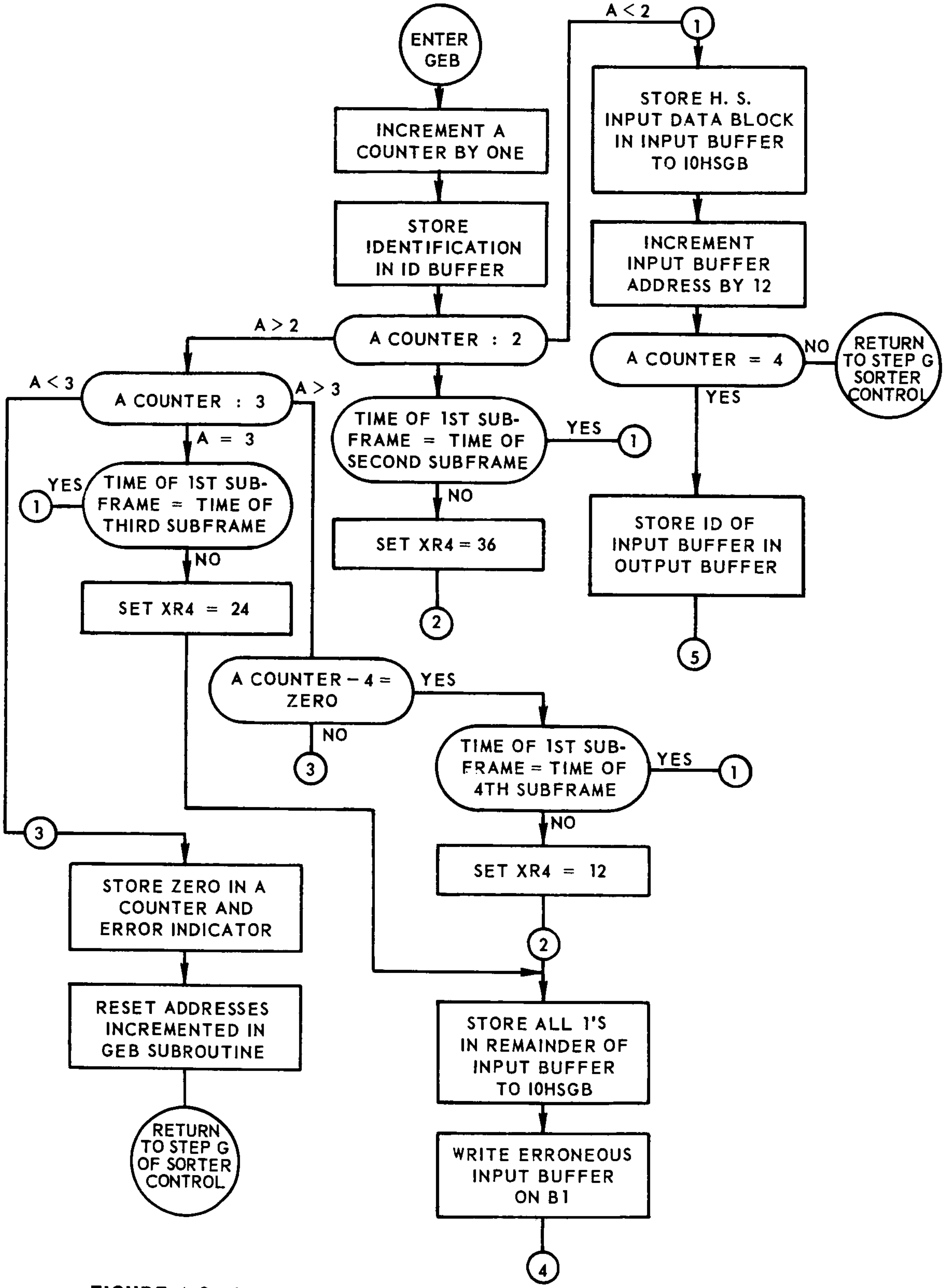


FIGURE 4-2. SUBCHANNEL 1 PROCESSING PROGRAM (GEB) *(Sheet 1 of 4)*

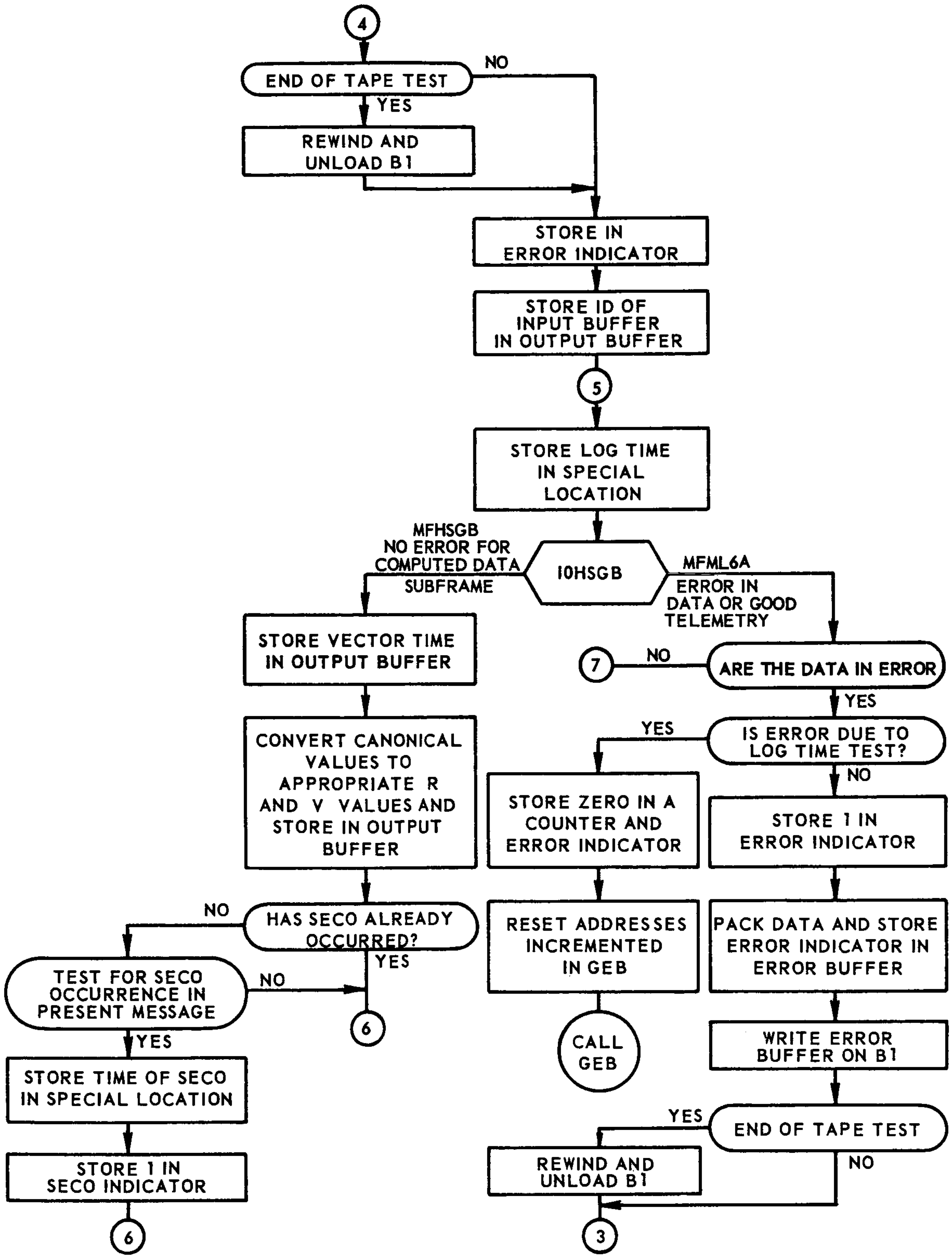


FIGURE 4-2. SUBCHANNEL 1 PROCESSING PROGRAM (GEB) *(Sheet 2 of 4)*

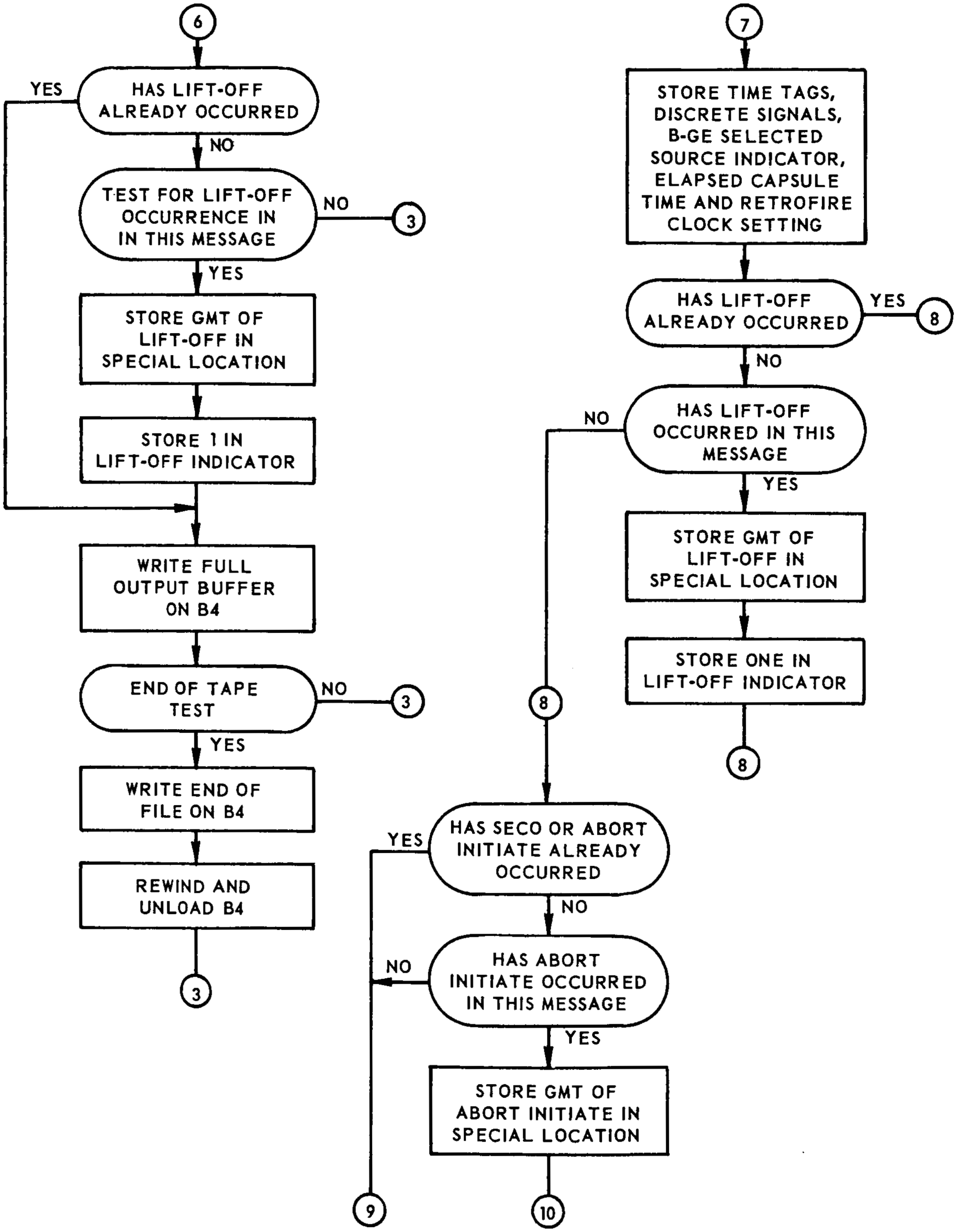


FIGURE 4-2. SUBCHANNEL 1 PROCESSING PROGRAM (GEB) *(Sheet 3 of 4)*

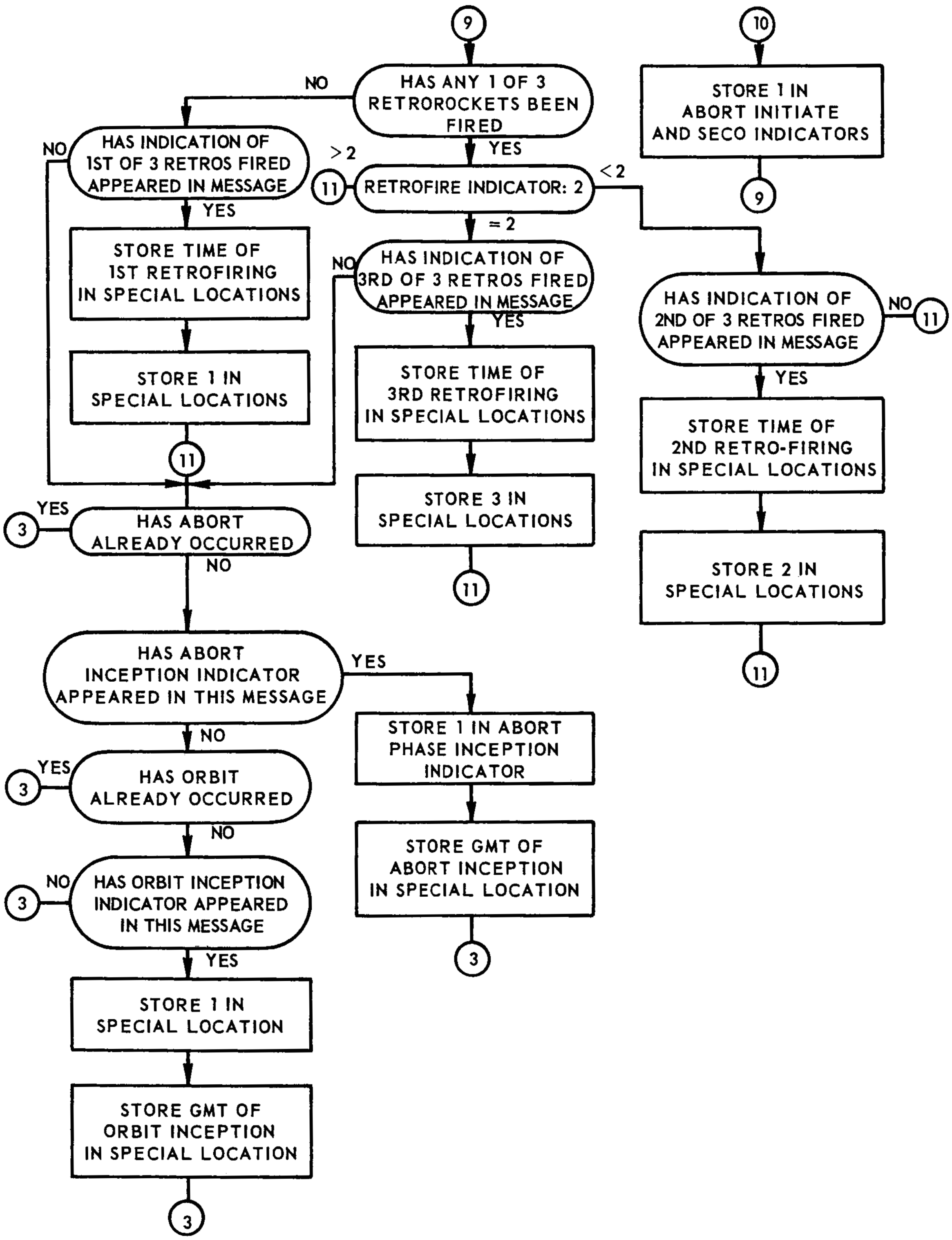


FIGURE 4-2. SUBCHANNEL 1 PROCESSING PROGRAM (GEB) *(Sheet 4 of 4)*

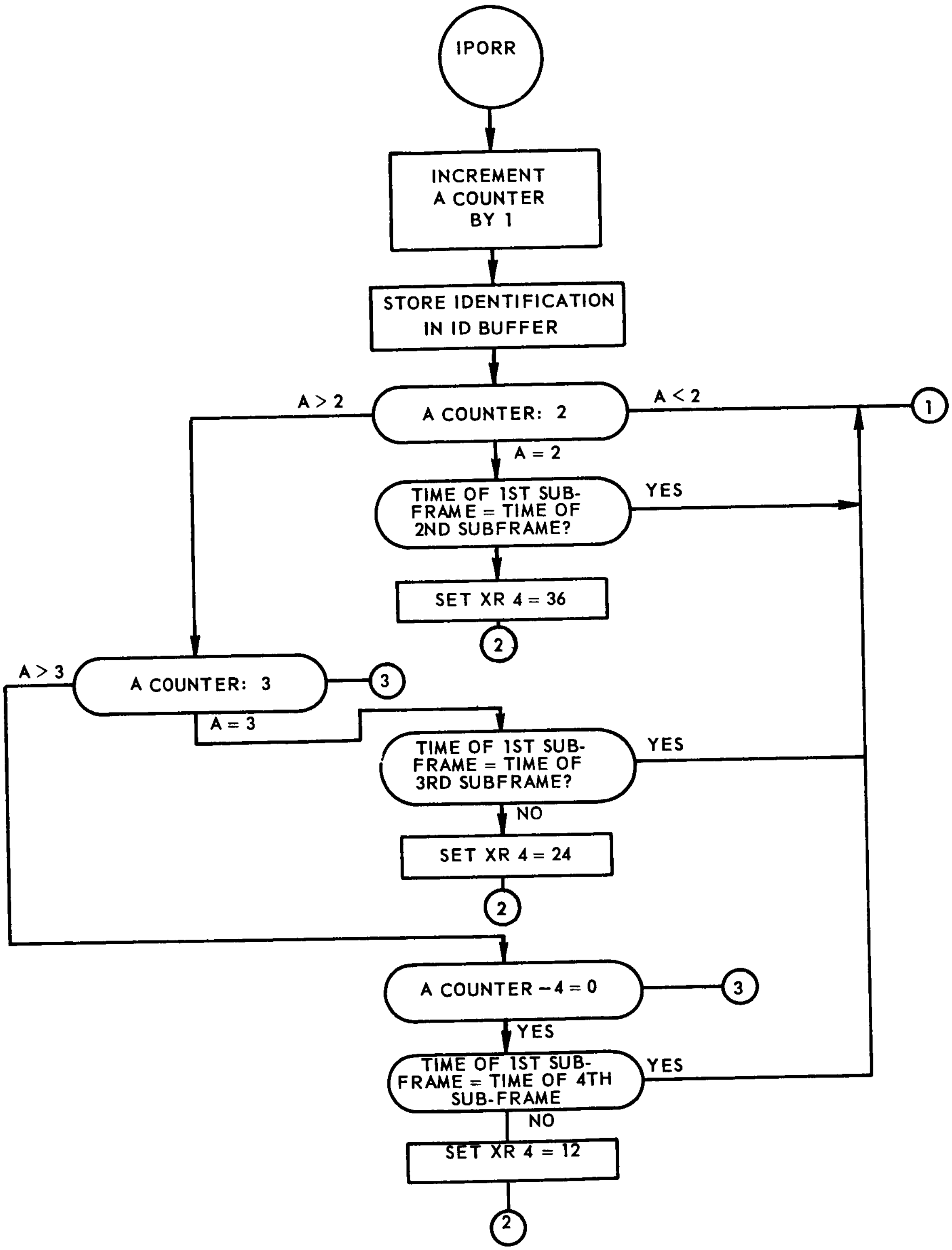


FIGURE 4-3. IP7090 HIGH-SPEED INPUT PROCESSOR PROGRAM (IPORR) *(Sheet 1 of 5)*

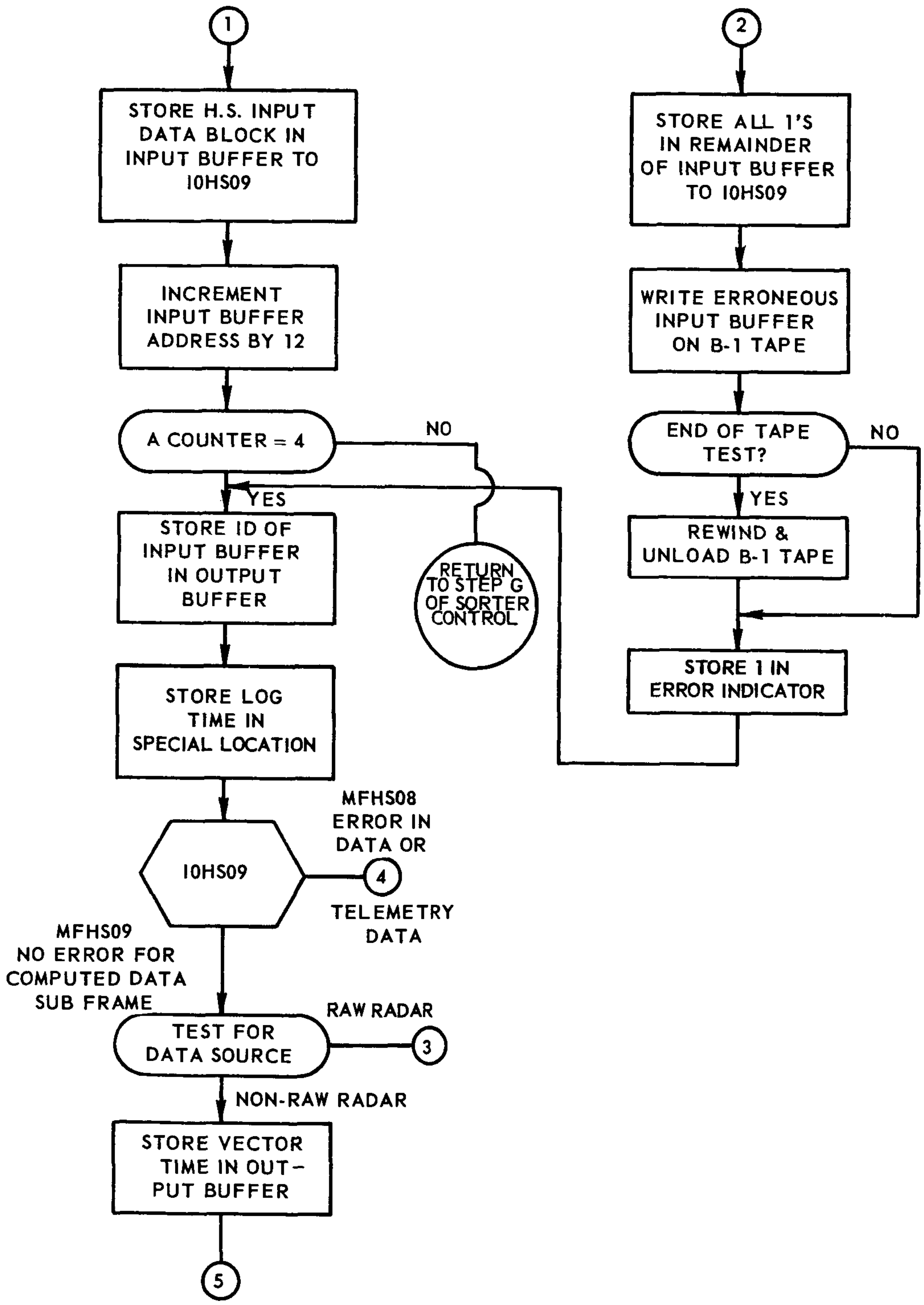


FIGURE 4-3. IP 7090 HIGH-SPEED INPUT PROCESSOR PROGRAM (IPORR) *(Sheet 2 of 5)*

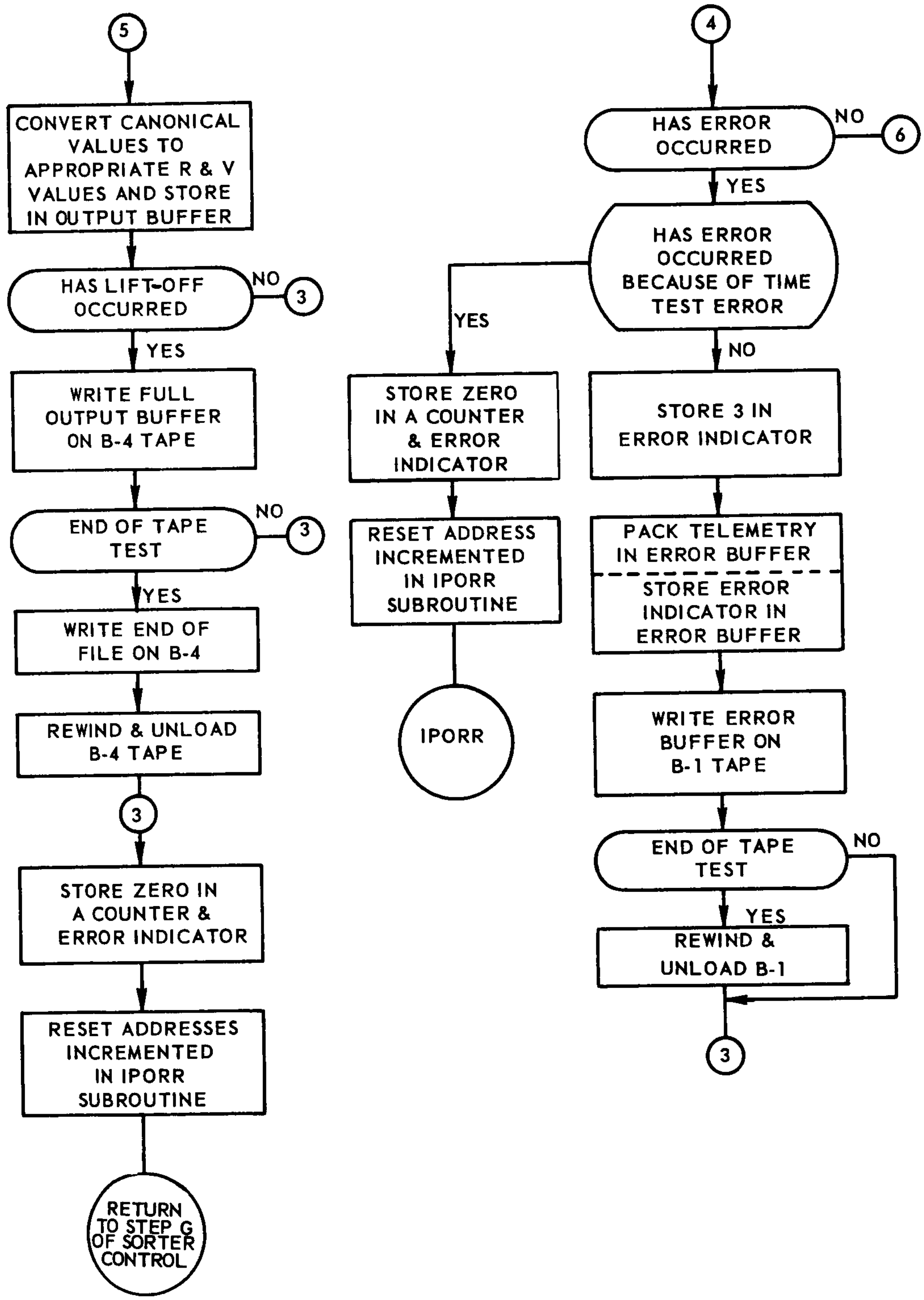


FIGURE 4-3. IP 7090 HIGH-SPEED INPUT PROCESSOR PROGRAM (IPORR) *(Sheet 3 of 5)*

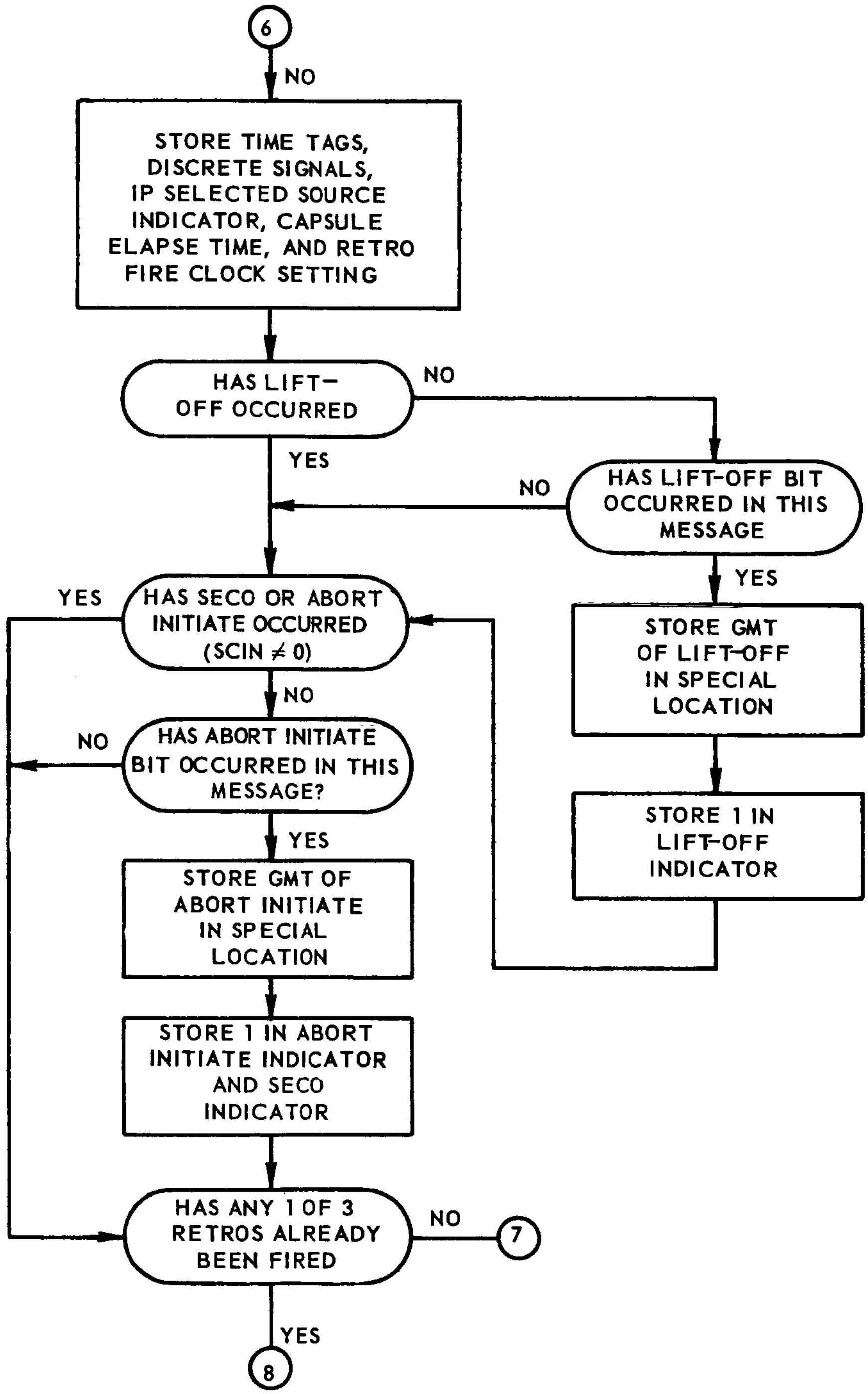


FIGURE 4-3. IP 7090 HIGH-SPEED INPUT PROCESSOR PROGRAM (IPORR) *(Sheet 4 of 5)*

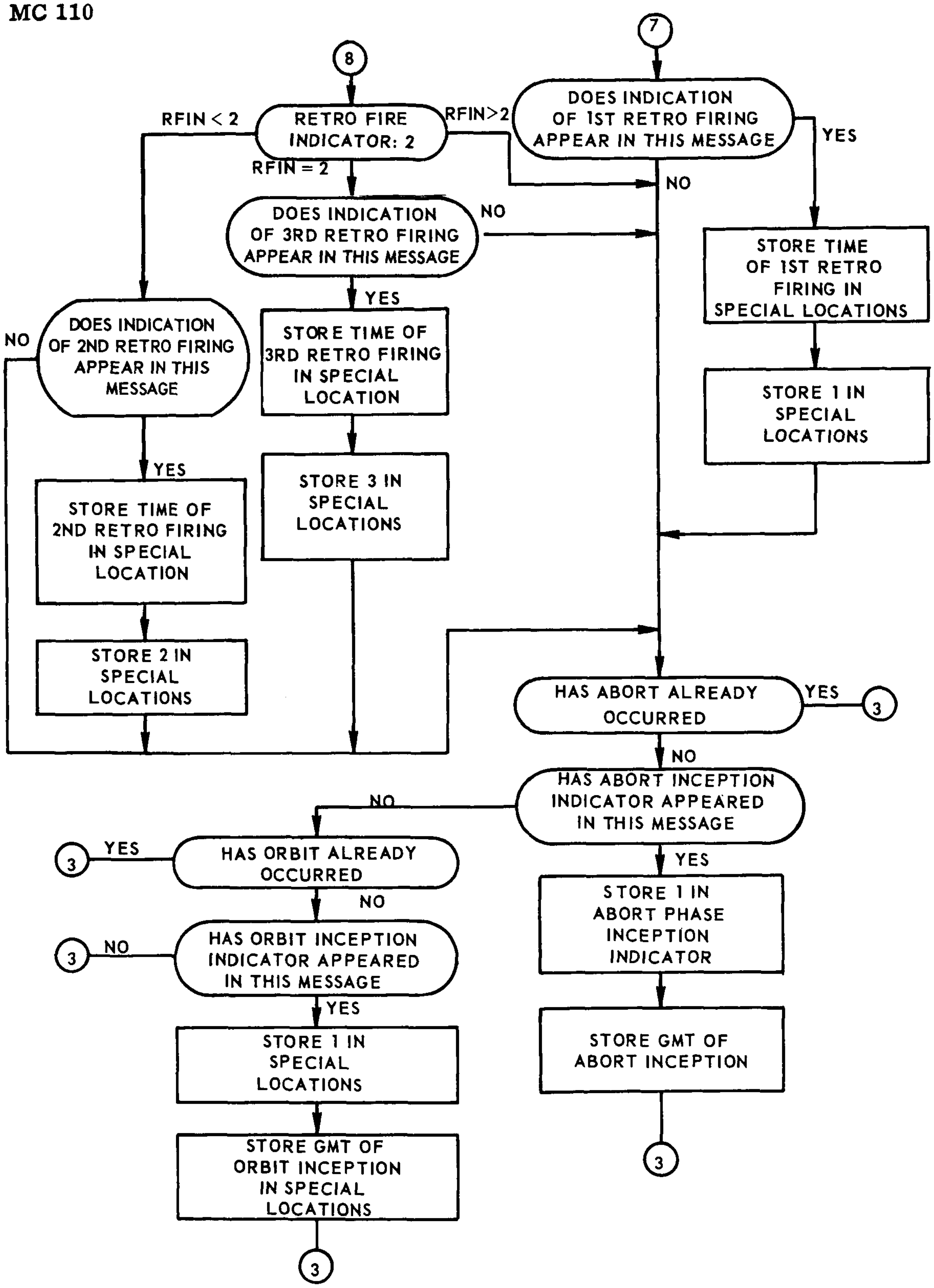


FIGURE 4-3. IP 7090 HIGH-SPEED INPUT PROCESSOR PROGRAM (IPORR) *(Sheet 5 of 5)*

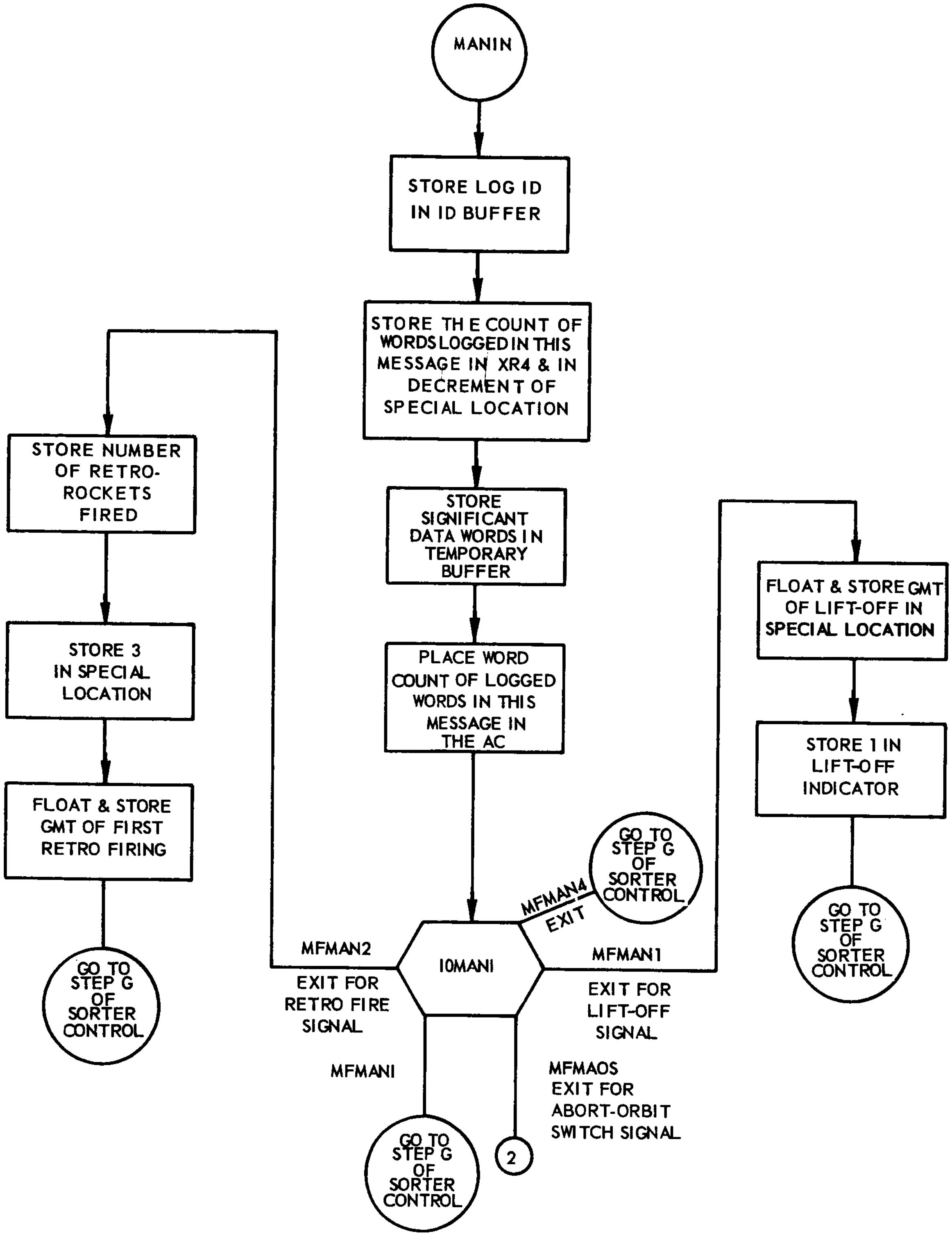


FIGURE 4-4. MANUAL INSERTION PROCESSOR PROGRAM (MANIN) *(Sheet 1 of 2)*

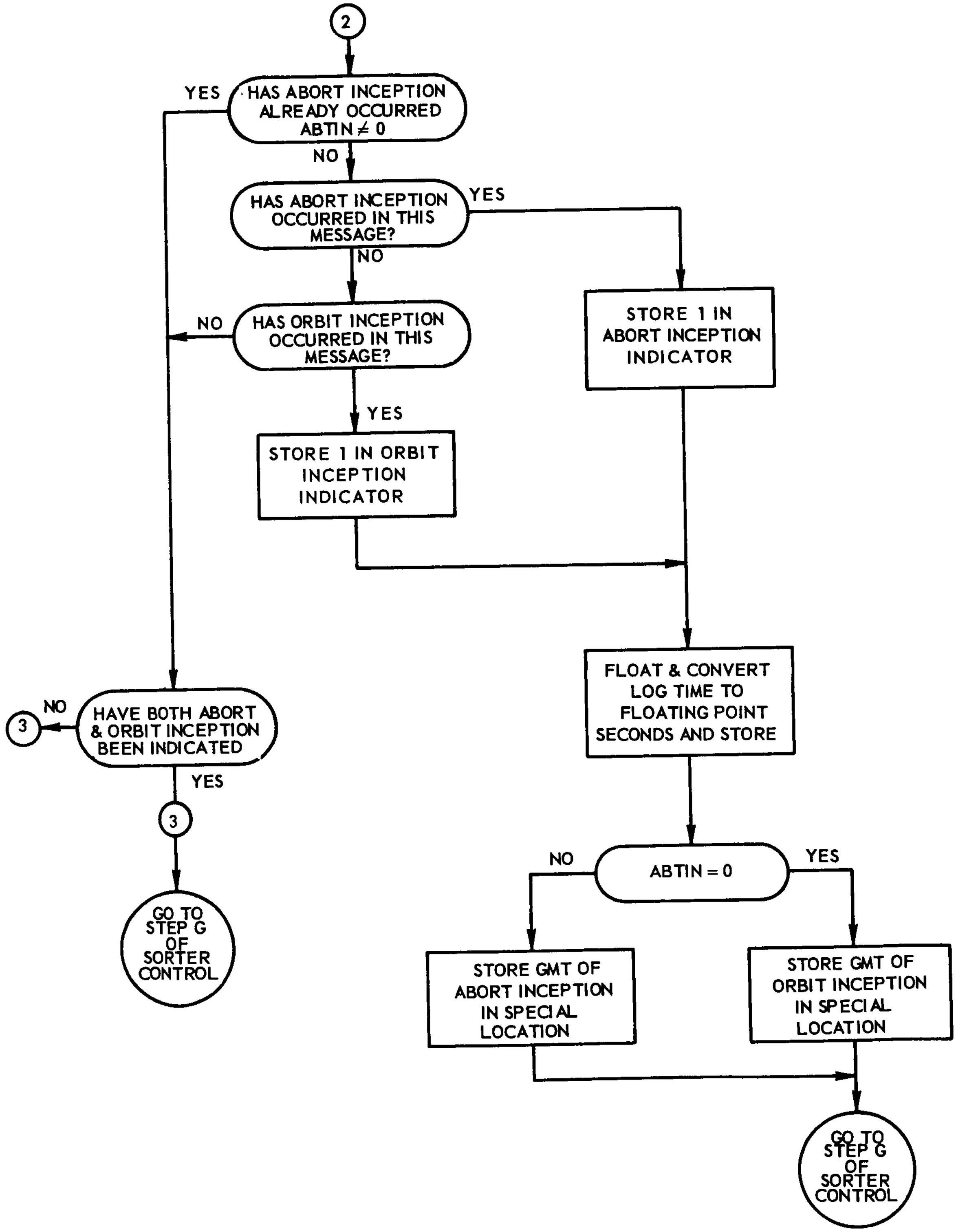


FIGURE 4-4. MANUAL INSERTION PROCESSOR PROGRAM (MANIN) *(Sheet 2 of 2)*

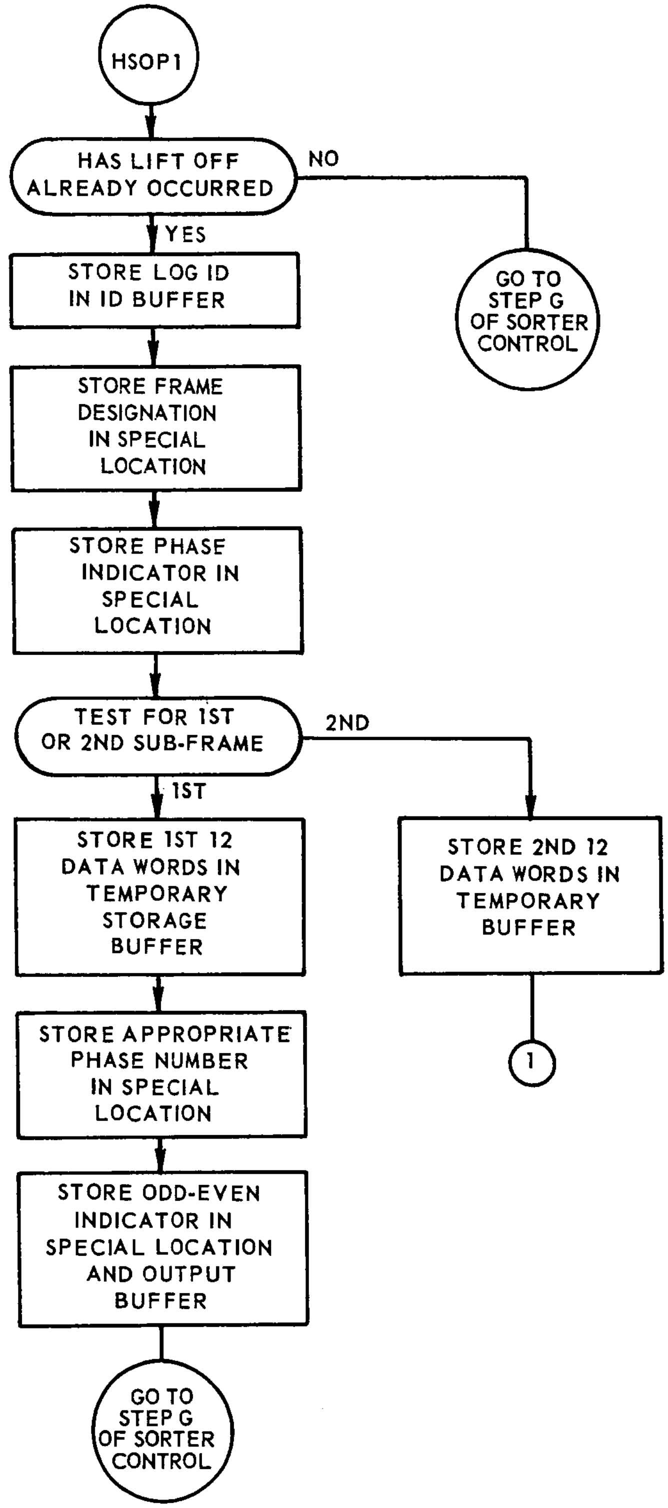


FIGURE 4-5. HIGH-SPEED OUTPUT TAPE WRITER PROGRAM (HSOP1) *(Sheet 1 of 8)*

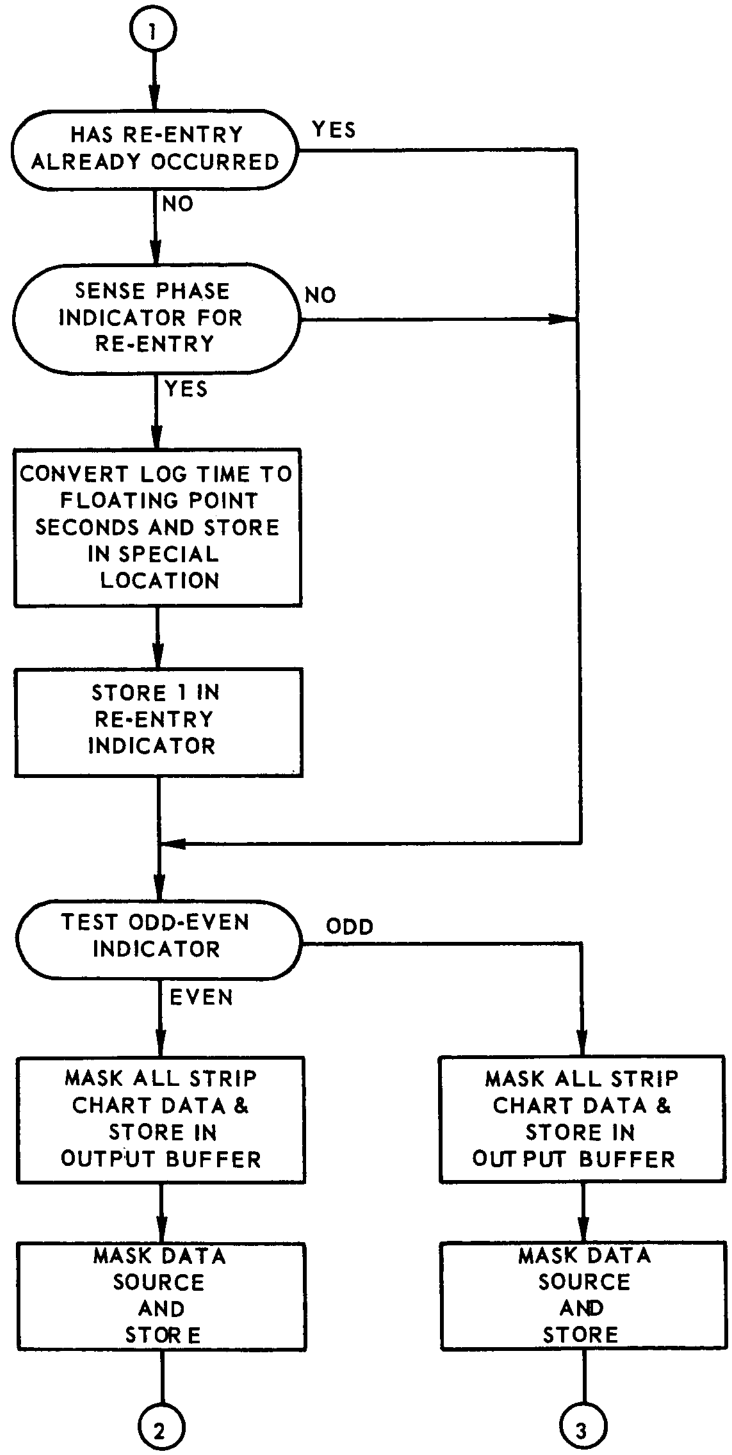


FIGURE 4-5. HIGH-SPEED OUTPUT TAPE WRITER PROGRAM (HSOP1) *(Sheet 2 of 8)*

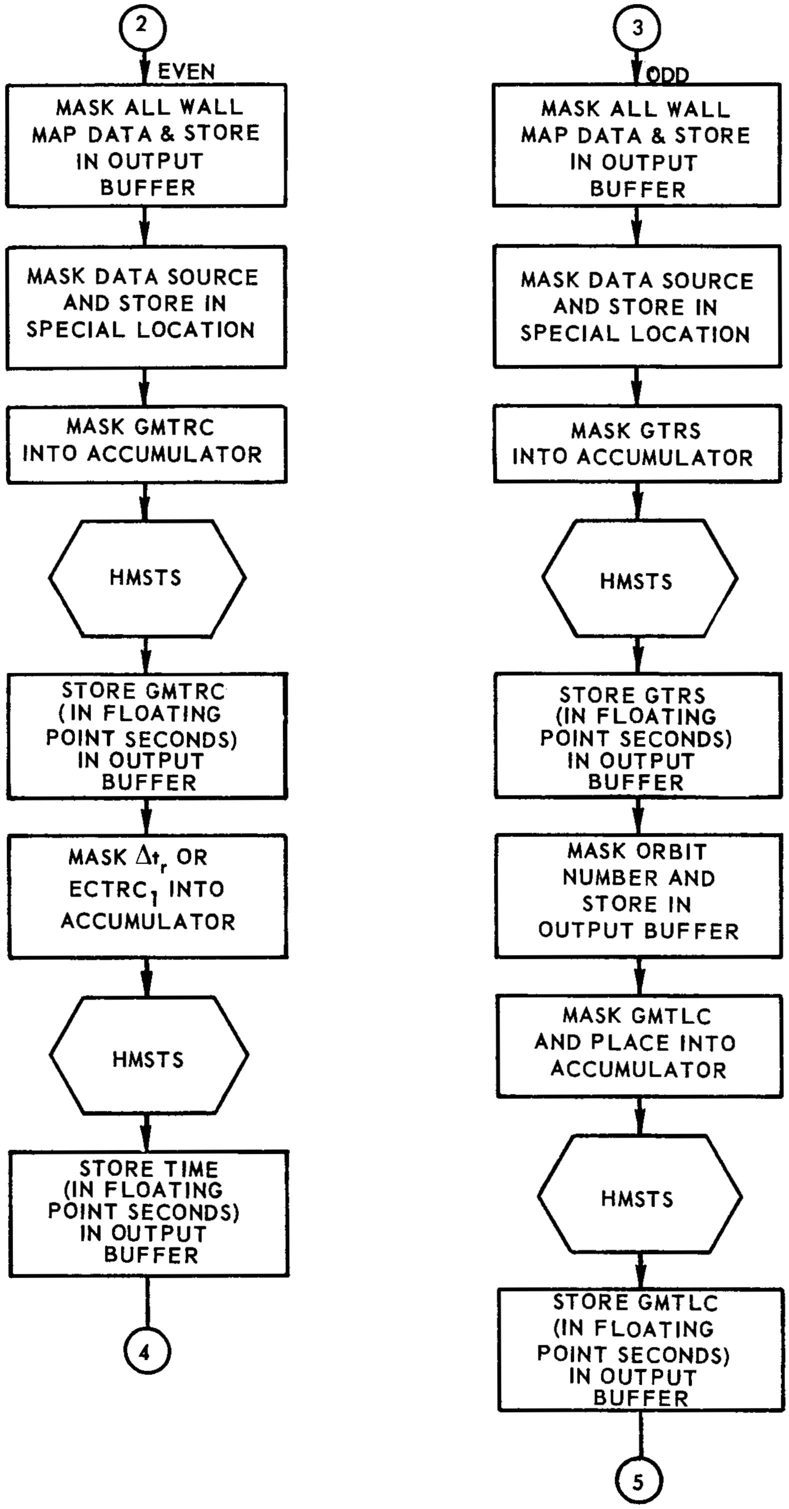


FIGURE 4-5. HIGH-SPEED OUTPUT TAPE WRITER PROGRAM (HSOP 1) *(Sheet 3 of 8)*

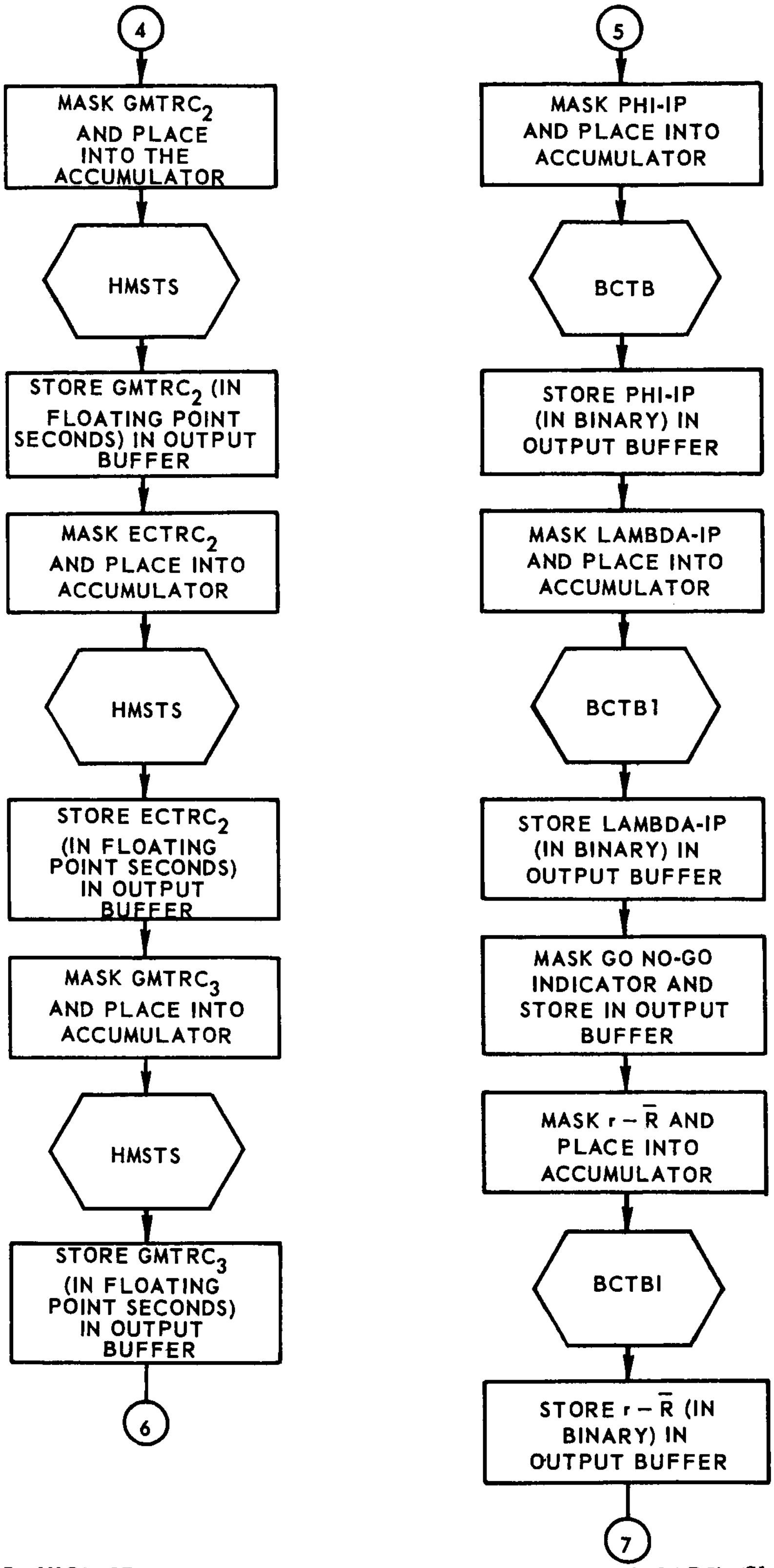


FIGURE 4-5. HIGH-SPEED OUTPUT TAPE WRITER PROGRAM (HSOP1) *(Sheet 4 of 8)*

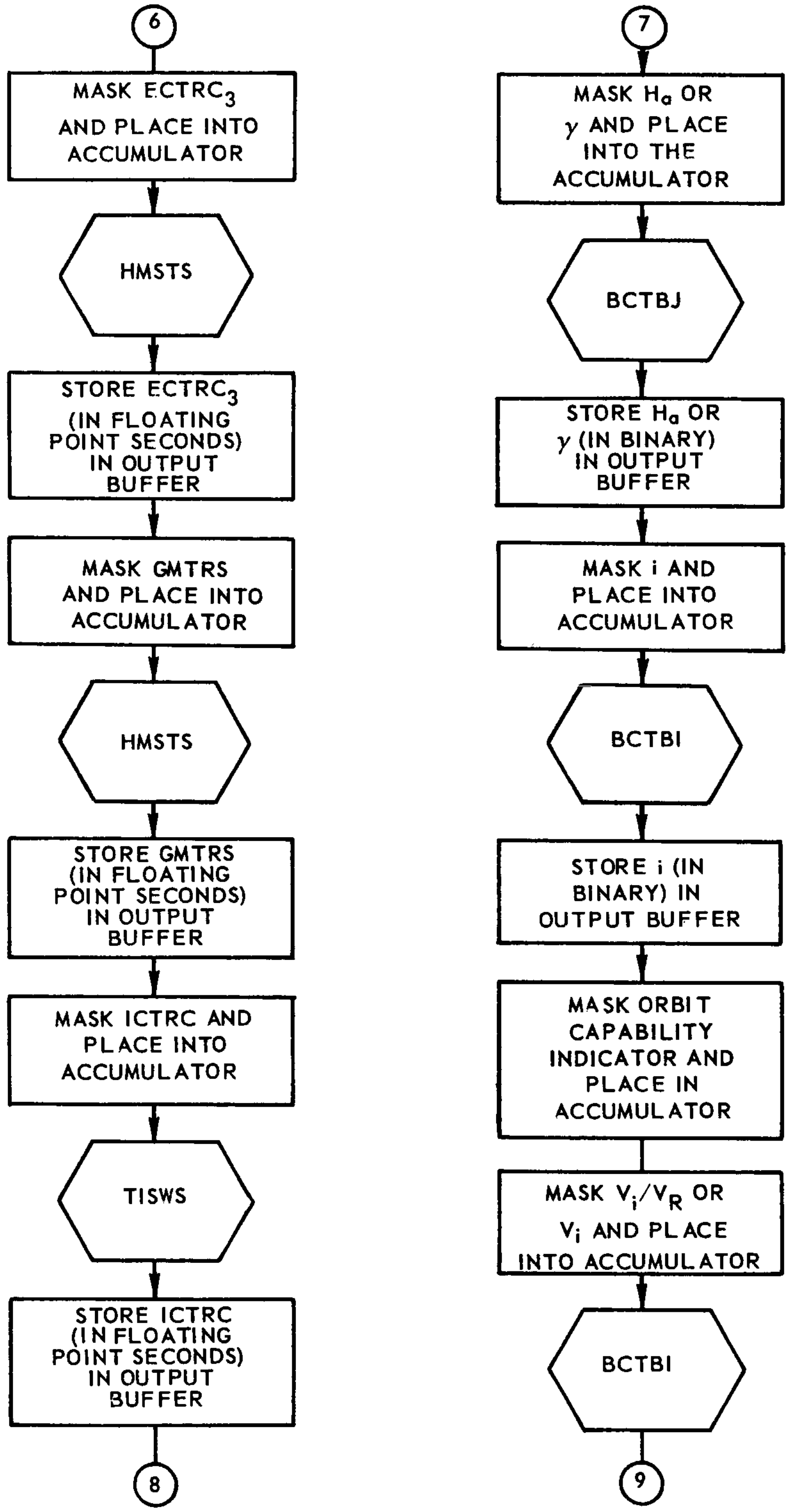


FIGURE 4-5. HIGH SPEED OUTPUT TAPE WRITER PROGRAM (HSOP1) *(Sheet 5 of 8)*

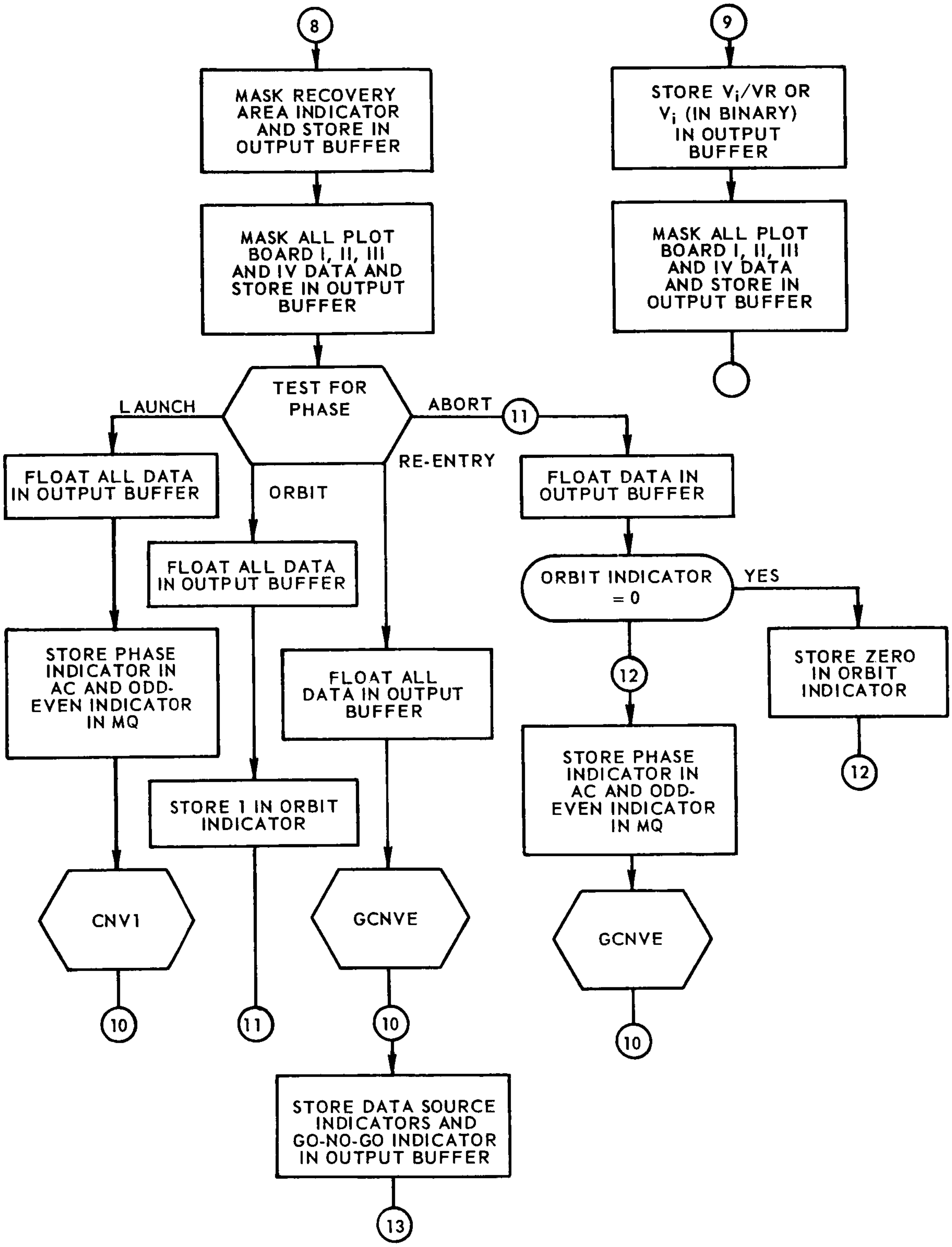


FIGURE 4-5. HIGH SPEED OUTPUT TAPE WRITER PROGRAM (HSOP1) *(Sheet 6 of 8)*

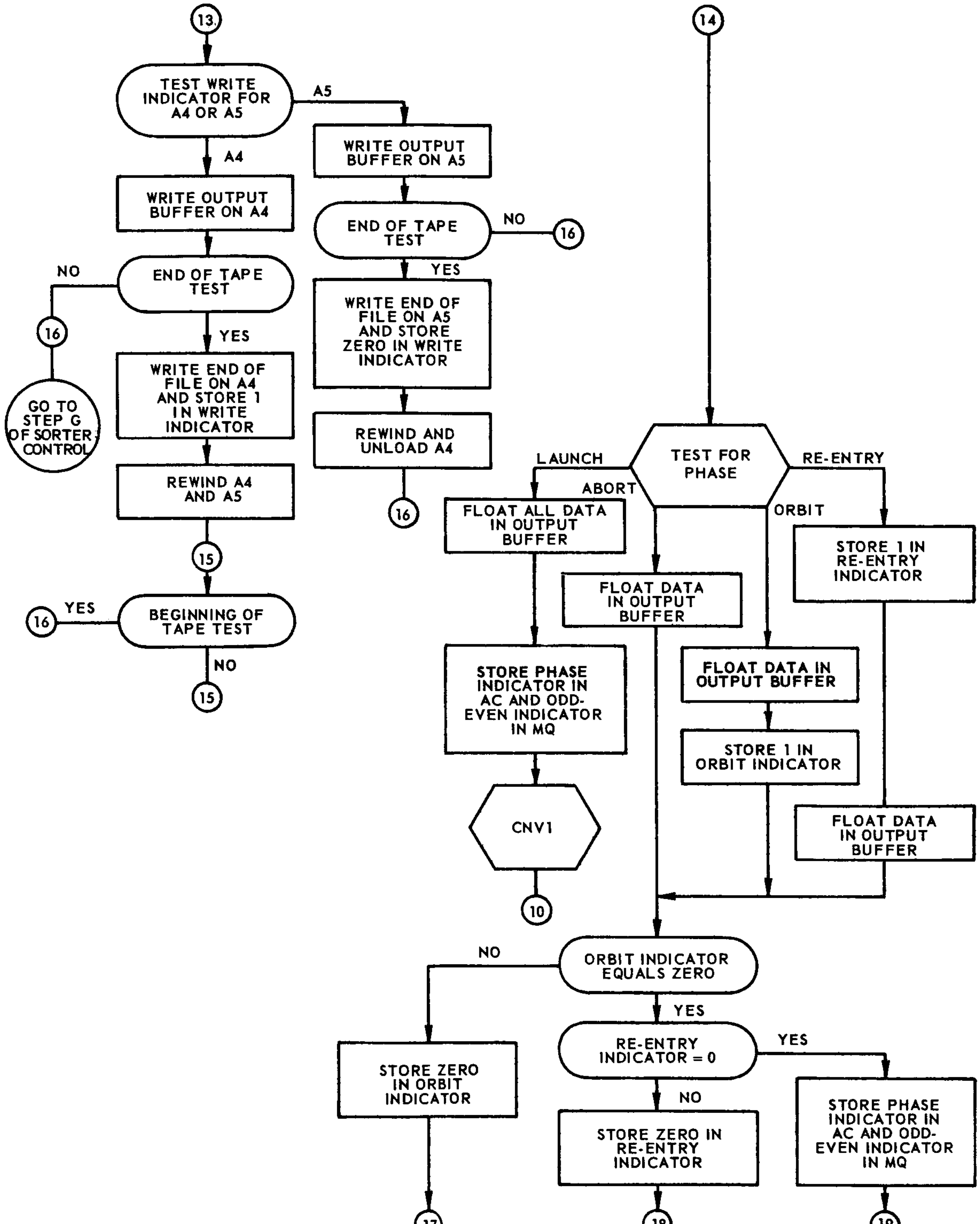


FIGURE 4-5. HIGH SPEED OUTPUT TAPE WRITER PROGRAM (HSOP1) *(Sheet 7 of 8)*

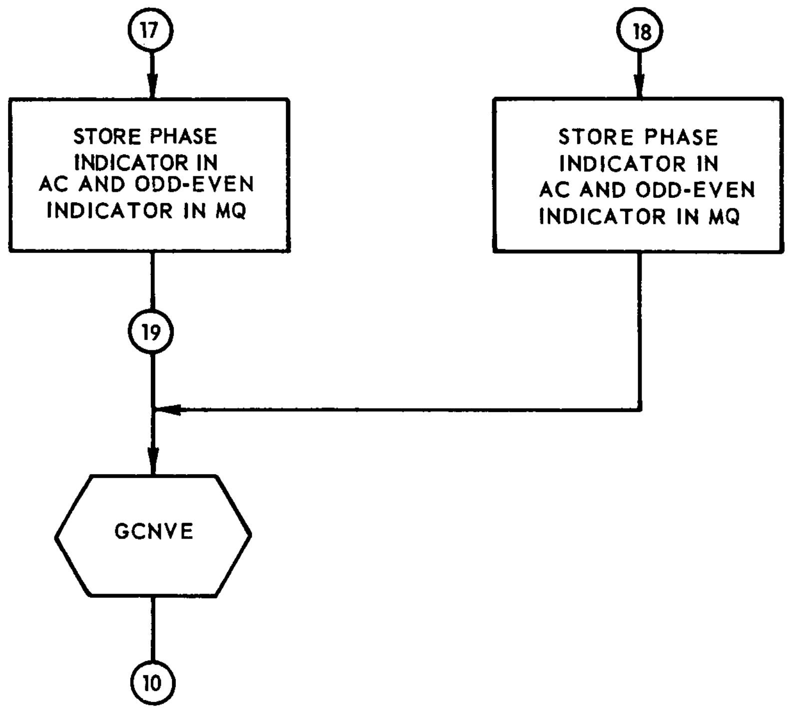


FIGURE 4-5. HIGH-SPEED OUTPUT TAPE WRITER PROGRAM (HSOP1) *(Sheet 8 of 8)*

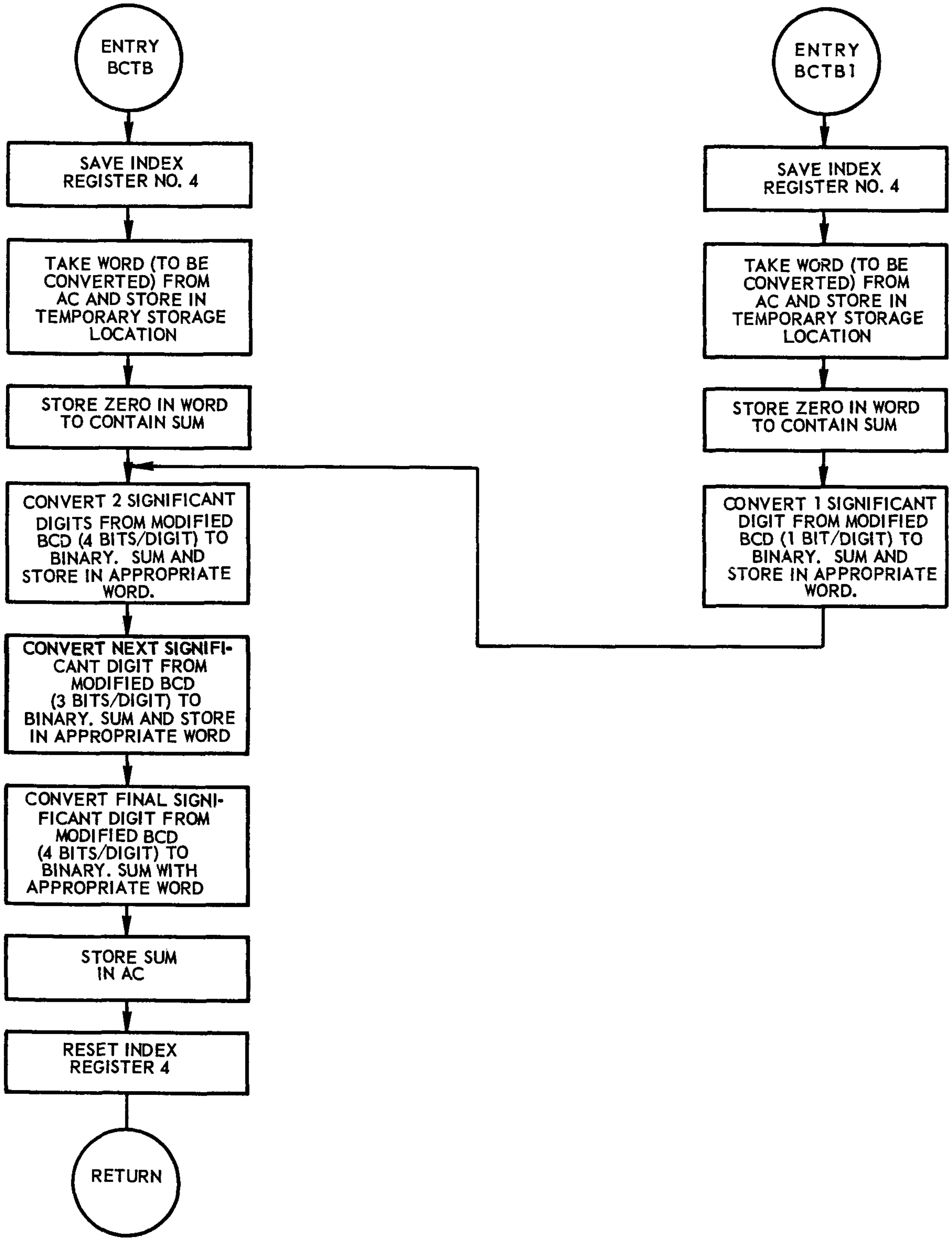


FIGURE 4-6. BCD WORD CONVERSION PROGRAM (A) (BCTB/BCTB1)

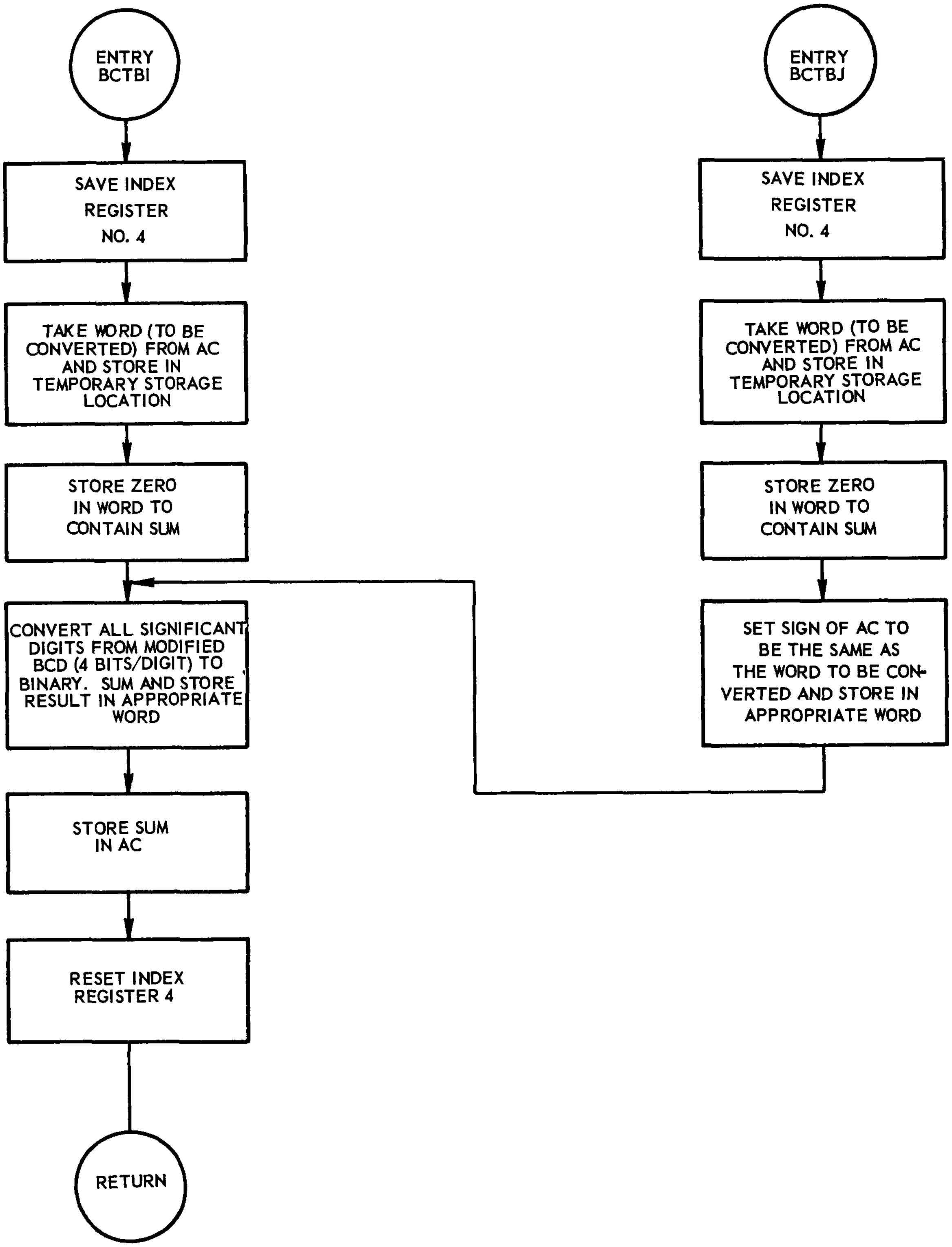


FIGURE 4-7. BCD WORD CONVERSION PROGRAM (B) (BCTBI/BCTBJ)

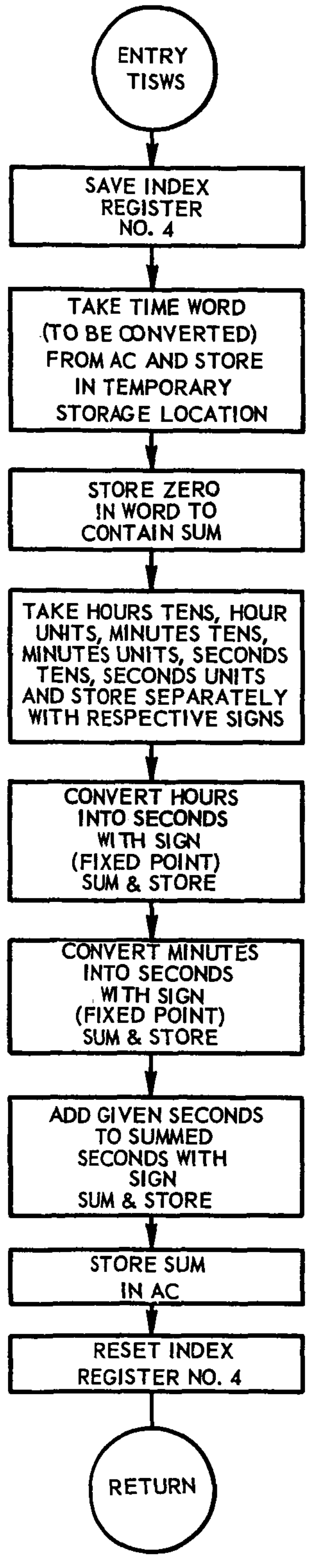


FIGURE 4-8. TIME WORD CONVERSION PROGRAM (A) (TISWS)

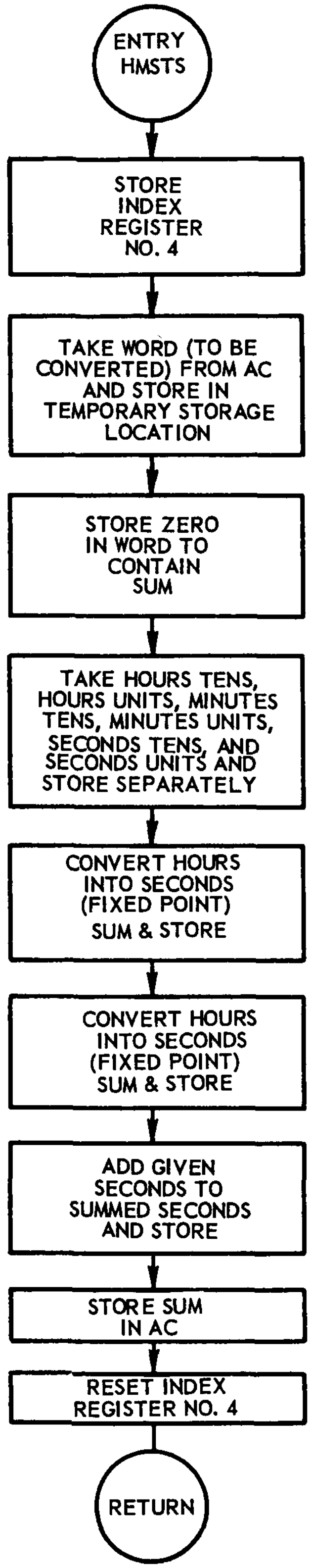


FIGURE 4-9. TIME WORD CONVERSION PROGRAM (B) (HMSTS)

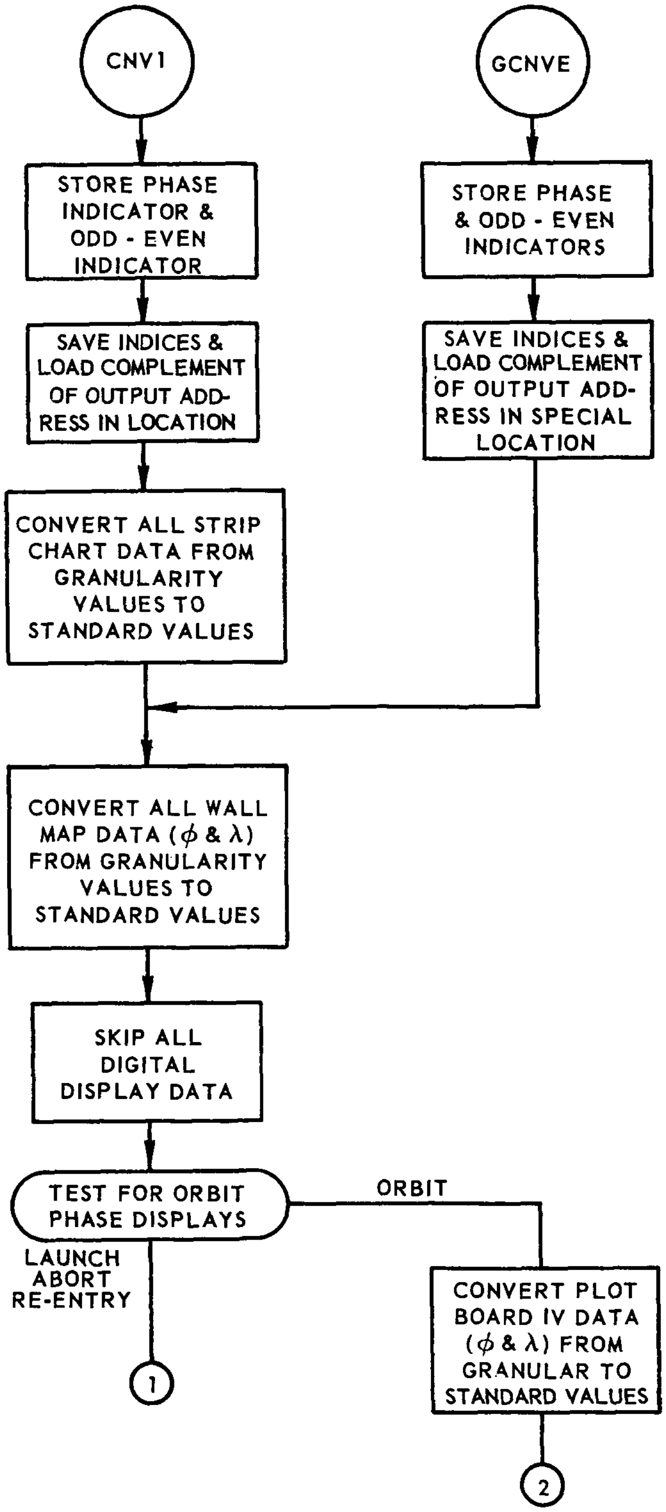


FIGURE 4-10. UNITS CONVERSION PROGRAM (CNV1/GCNVE) *(Sheet 1 of 3)*

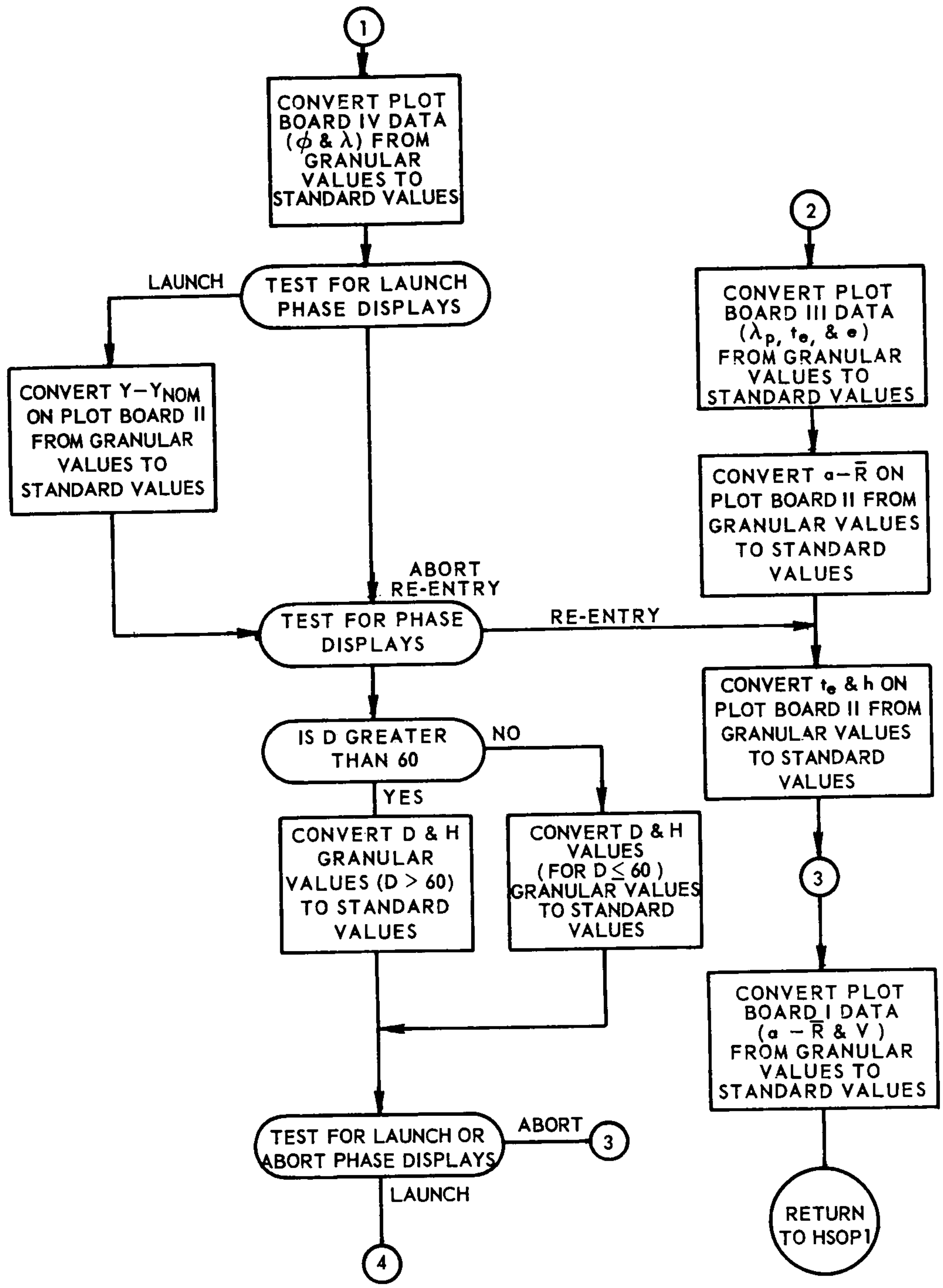


FIGURE 4-10. UNITS CONVERSION PROGRAM (CNV1/GCNVE) *(Sheet 2 of 3)*

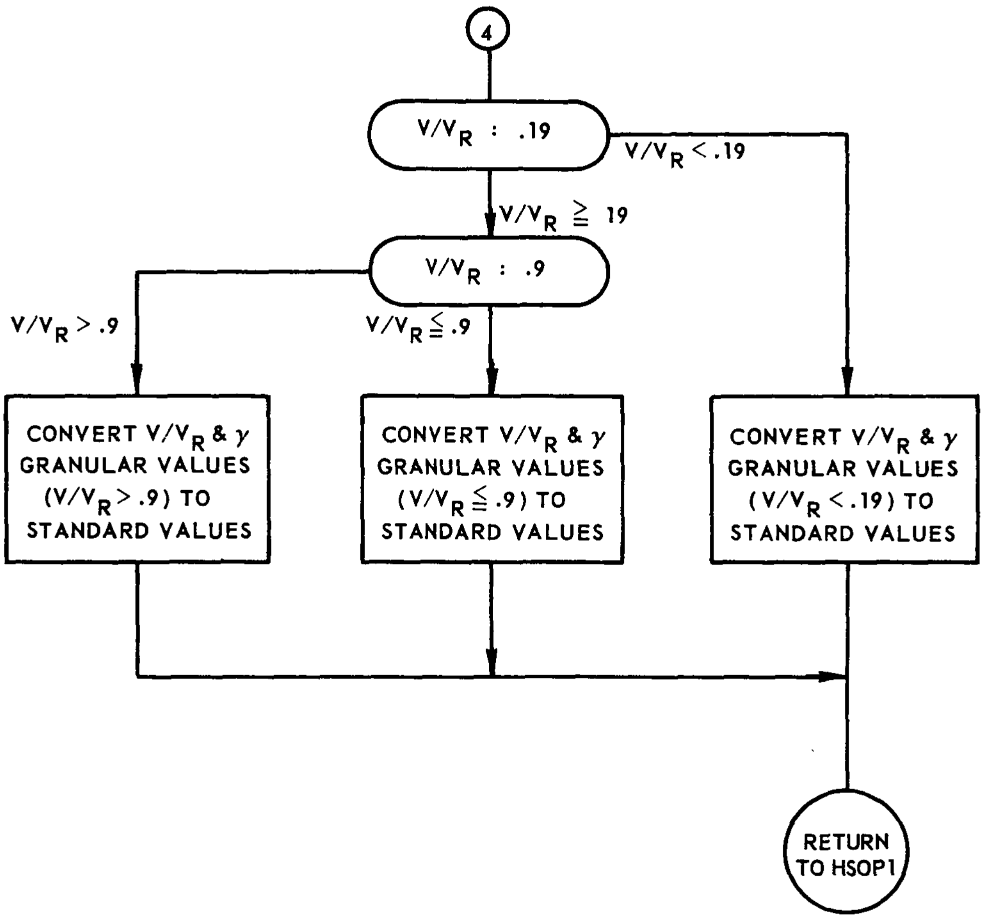


FIGURE 4-10. UNITS CONVERSION PROGRAM (CNV1/GCNVE) *(Sheet 3 of 3)*

4.2 DISCRETE EVENT PROCESSOR PROGRAM (GETME)

GETME searches the second file of the high-speed input tape (B4) created by SORTER from data on the Mercury Program System log tape. This second file contains a record of the occurrence of seven discrete events. The information located on these events is placed into COMMON storage for reference by Monitor. However, the Postflight Reporter Program cannot continue beyond the GETME program if the first word of any record in this second file, the GMT of lift-off, is not found. The flow chart for GETME is shown in Figure 4-11.

4.2.1 Input Requirements

The input to GETME is the second file of the B4 tape created by the SORTER program. This file contains 1,000 copies of a record containing all data pertaining to the occurrence of seven discrete events (see Subsection 4.2.2).

4.2.2 Output Requirements

GETME outputs the following inormation about the discrete events:

Greenwich Mean Time of lift-off (GMTLO)

Number of retrorockets fired

Elapsed time from lift-off to sustainer engine cut-off (SECO)

Time of abort inception referenced to lift-off

Time of orbit inception referenced to lift-off

Time of re-entry inception referenced to lift-off

Time of firing first retrorocket referenced to lift-off

This information is placed into COMMON storage before the exit of this program.

4.2.3 Method

This program skips the first file on the high-speed input tape (B4) and searches 1,000 records of the second file for the data pertaining to the discrete events. If none of the records is accepted, the program places a one (1), in the error indicator (see Subsection 4.2.4). Another attempt may be made at obtaining the data.

If one of the records is accepted, GETME tests for the existence of the events within the record. If an event did not occur, the particular location for that event is supplied with a minus one (-1). All data is then transferred to COMMON storage, and all times are referenced to GMT of lift-off (GMTLO).

4.2.4 Usage

Call Statement

CALL GETME (NOYES)

NOYES is an error indicator.

1 indicates an error in tape setup

2 indicates normal return

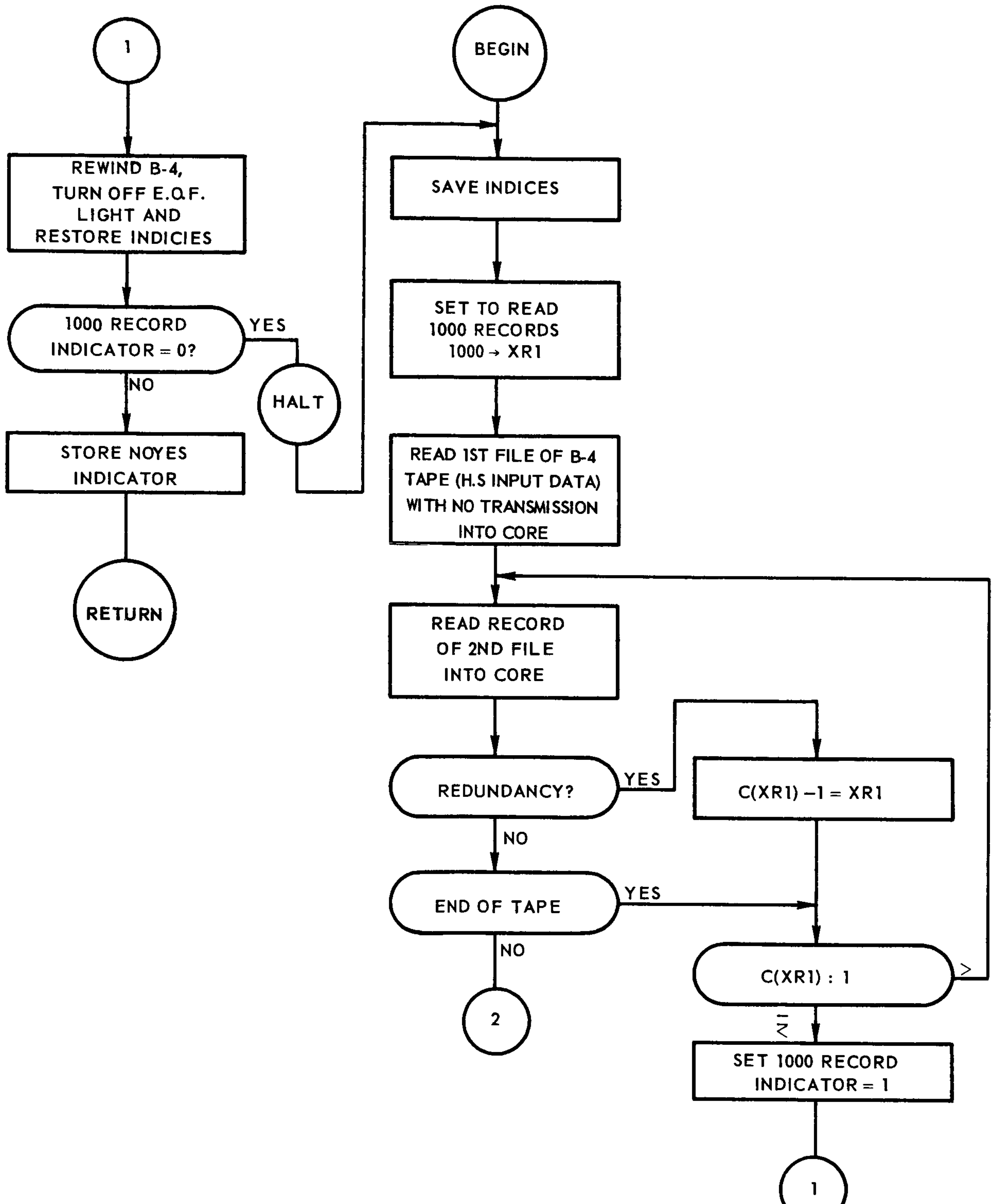


FIGURE 4-11. DISCRETE EVENT PROCESSOR PROGRAM (GETME) *(Sheet 1 of 2)*

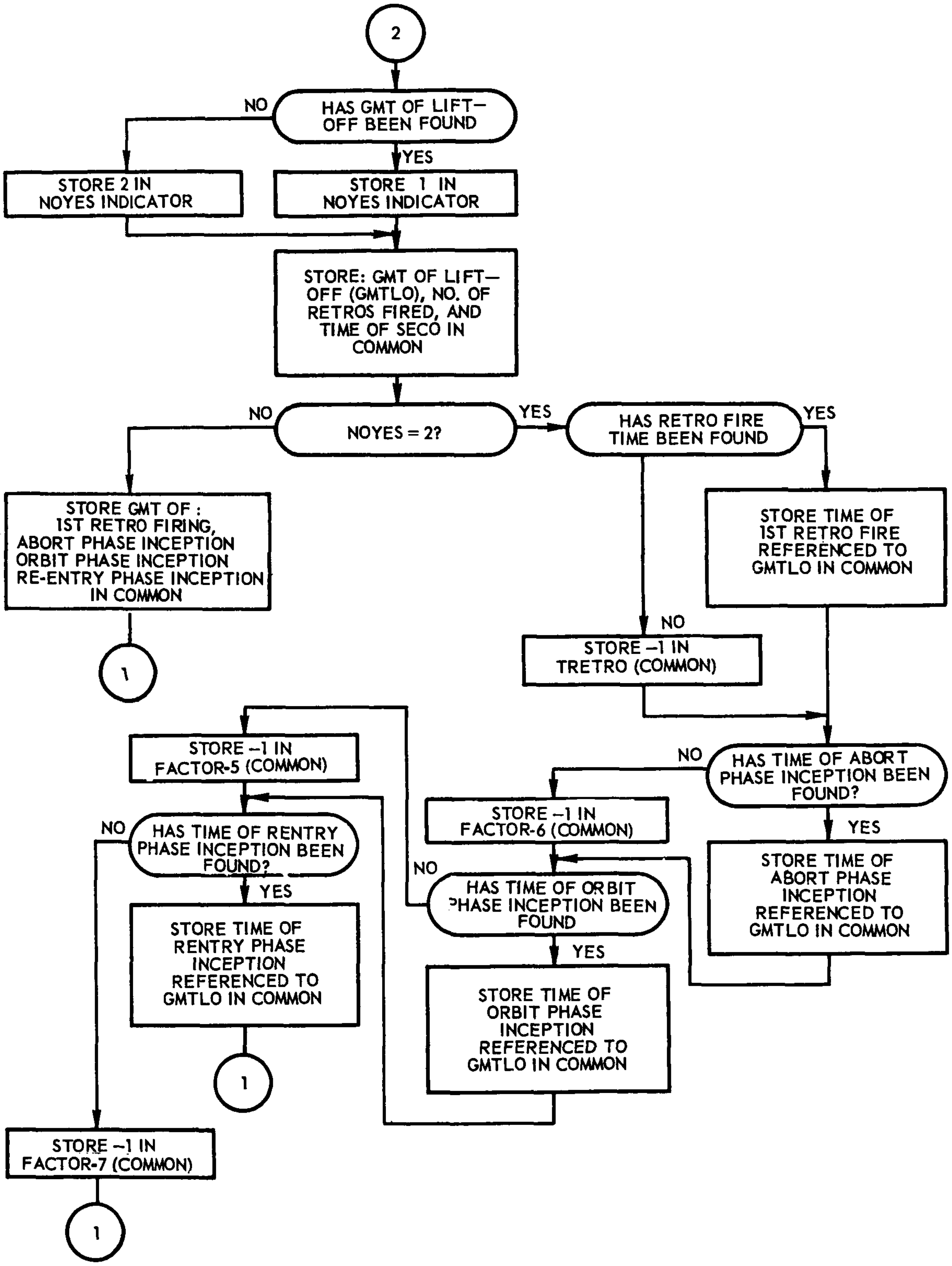


FIGURE 4-11. DISCRETE EVENT PROCESSOR PROGRAM (GETME) *(Sheet 2 of 2)*

4.3 HIGH-SPEED OUTPUT PROCESSOR PROGRAM (DONOUT)

DONOUT processes the high-speed output tape (A4), which contains the data removed from the Mercury Program System log tape and unpacked by SORTER. DONOUT removes the data from the A4 tape according to a predetermined time and flight phase. It places available data into locations of COMMON storage corresponding to the specific high-speed output quantities. The flow chart for DONOUT is shown in Figure 4-12.

4.3.1 Input Requirements

The input to this program is high-speed output data written on the A4 tape (and A5 tape if enough room is not available on a single tape). The data is in 44-word records. These records consist of 10 words of heading and 34 words of decoded data. At the physical end of an A4 or A5 tape is a single end of file. At the logical end of all tapes are two ends of file.

A closed subroutine called BRINR is also used within DONOUT.

4.3.2 Output Requirements

DONOUT places into COMMON storage the high-speed output messages that drove the displays at Cape Canaveral. These messages are obtained according to a unique time tag. In the case of processed information, the time tag is the vector time associated with the input message which gave rise to the output. If such a vector time exists, the data source and vector time are output. If the vector time does not exist, an error return is indicated. Then, if no processed information is available, log time is used instead of vector time. (In the case of raw radar, a processor is not yet written and there is no available logic concerning it.)

4.3.3 Method

DONOUT reads a high-speed output block from the A4 tape. The phase (Launch, Abort, Orbit or Re-entry) is determined and compared with the phase requested in the call statement (see Subsection 4.3.4). If the phase of the block is numerically less than the phase requested, the phase requested is further down the tape and another record is read. If the phase of the block is greater than the phase requested and if the requested phase is different from the one previously requested, then the A4 tape is rewound so that a new record may be read for the new phase.

If the phase of the high-speed output block is the same as the one requested, the time of the output message is examined. If the time of the message is less than the time requested, the next record is read. If the time of the message is greater than the time requested, the program backspaces until a message of the correct time and phase is determined. If the time requested for the particular phase does not exist within the phase, an error is indicated.

Should the tape contain data from one phase, receive data from a second phase and subsequently have more data from the original phase, the program backspaces the tape to determine whether there have been two changes of phase. If possible, DONOUT will locate the last message in the proper phase and present it for decoding.

Whenever records are read from the A4 tape, tests are made for redundancies and ends of file. In order to read the next record, a closed subroutine called BRINR is used within DONOUT. This routine reads in the next record, examines for redundancy and end of file, and then exits. If redundancies exist, backspacing action is taken to reread these records. If the records can be reread, they are used; if they cannot be reread, the tape spaces ahead to the next record.

If an end of file on A4 or A5 exists, the tape is read a second time to determine whether a second end of file is on it. If so, this is the end of the data and an exit is made. If only one end of file exists on the tape, the physical end of the tape is reached and the other tape should be used. The program has the capacity to switch back and forth between A4 and A5 until enough tapes have been read to read in all the data.

4.3.4 Usage

Call Statement

CALL DONOUT (TEMPO1, J23, JERROR, NDATA)

TEMPO1 is a time indicator containing the requested vector time.

J23 is a phase indicator containing one of the following

1 indicates launch phase (MA or MR)

2 indicates abort phase (MR)

3 indicates orbit phase (MA)

4 indicates re-entry phase (MA)

5 indicates high or medium abort phase (MA)

6 indicates low abort phase (MA)

JERROR is an error indicator containing one of the following:

1 indicates no more output on the log tape for the requested phase

2 indicates output log but no transmission through the DCC

3 indicates normal return and normal transmission

4 indicates redundancy

NDATA is a data source indicator

1 indicates IP 7090 data

2 indicates B-GE data

3 indicates raw radar data

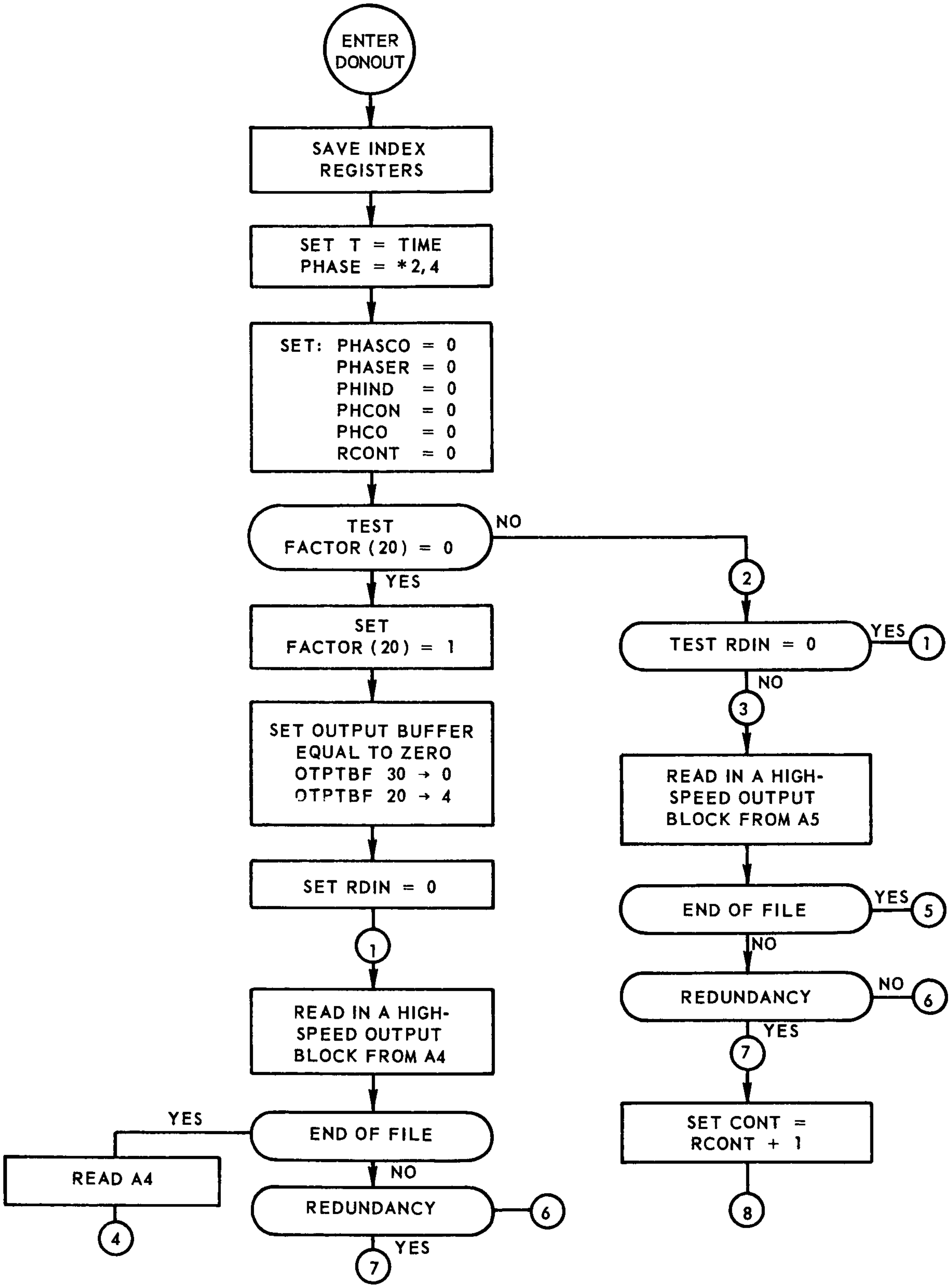


FIGURE 4-12. HIGH-SPEED OUTPUT PROCESSOR PROGRAM (DONOUT) *(Sheet 1 of 10)*

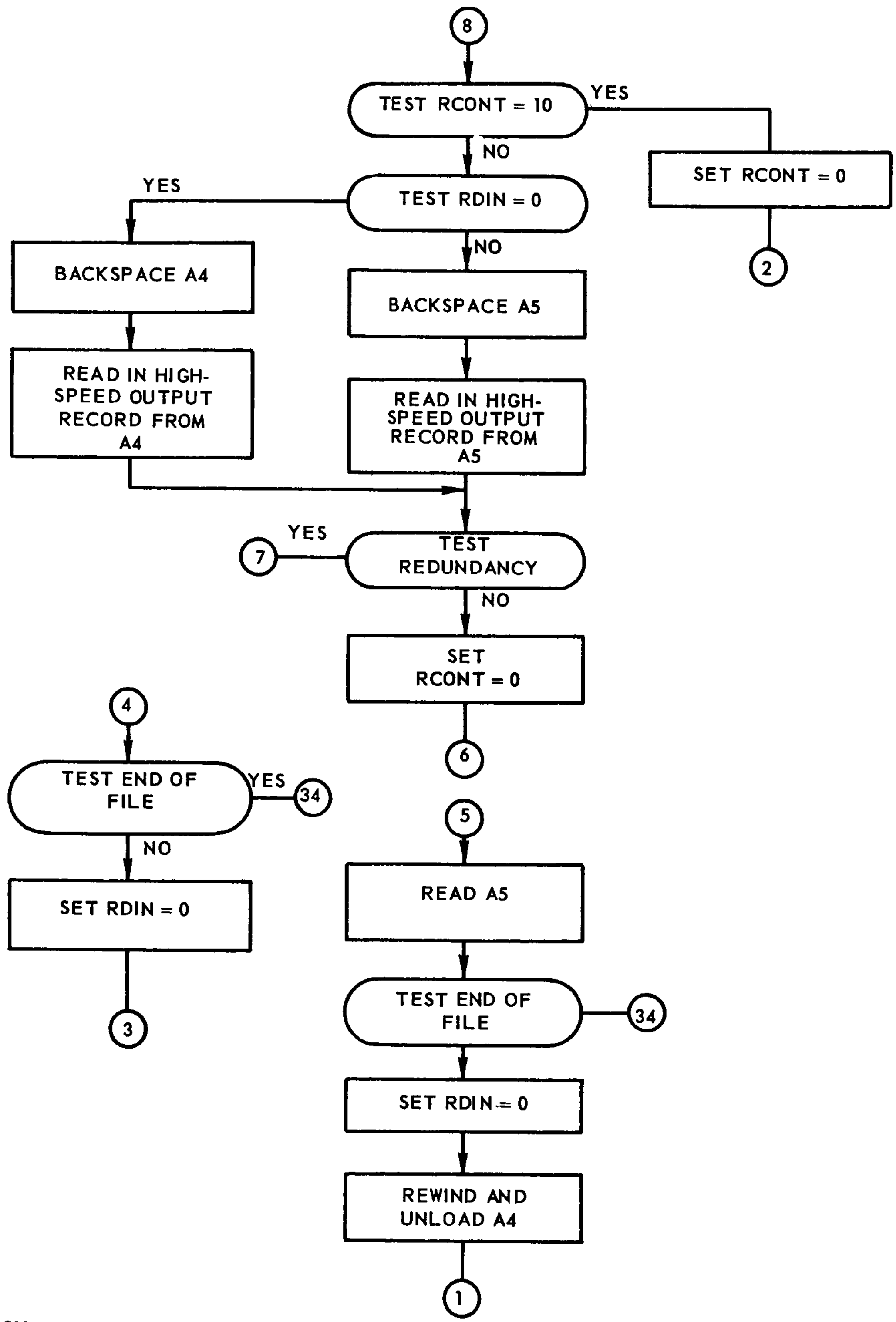


FIGURE 4-12. HIGH SPEED OUTPUT PROCESSOR PROGRAM (DONOUT) *(Sheet 2 of 10)*

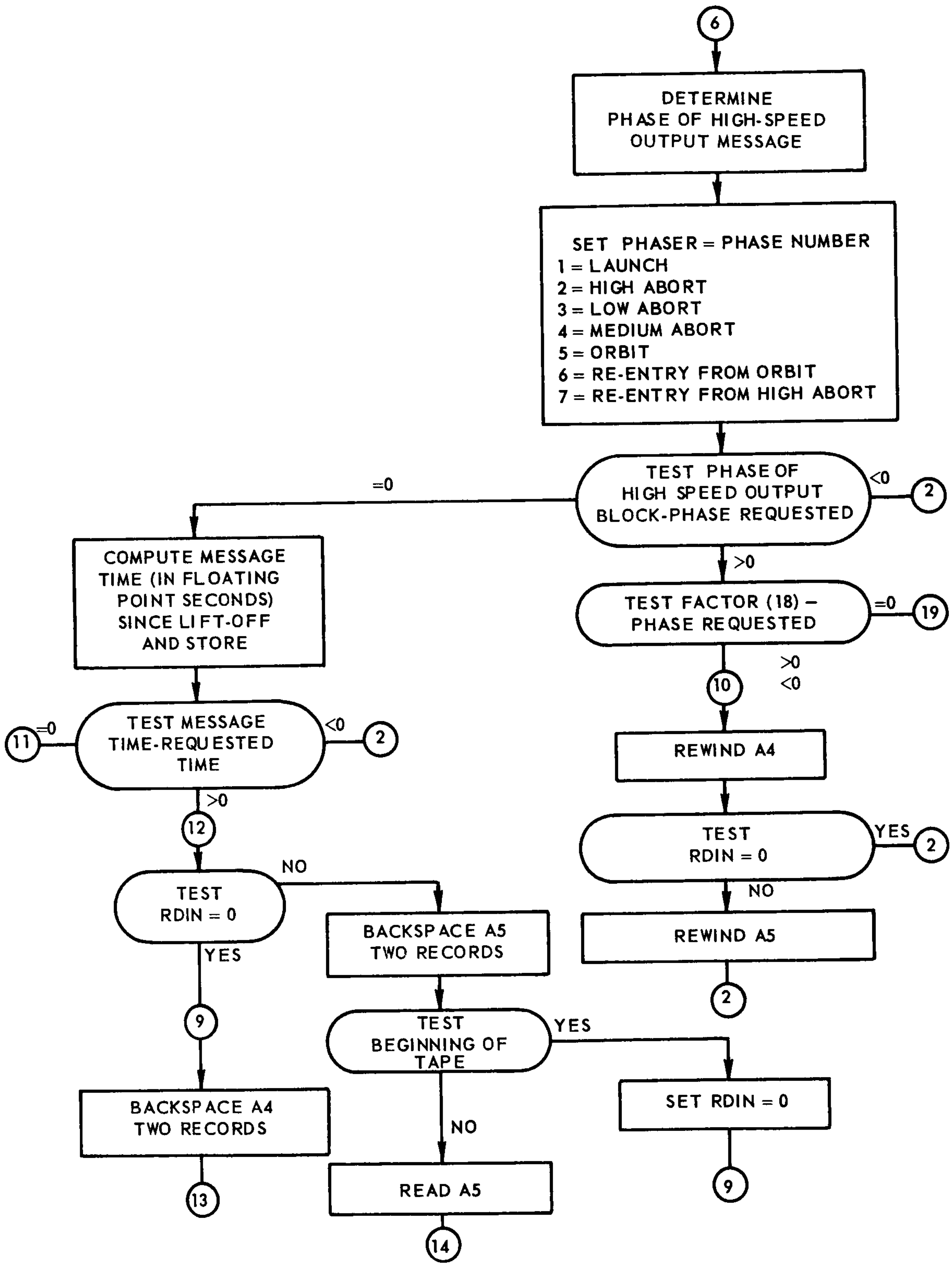


FIGURE 4-12. HIGH SPEED OUTPUT PROCESSOR PROGRAM (DONOUT) *(Sheet 3 of 10)*

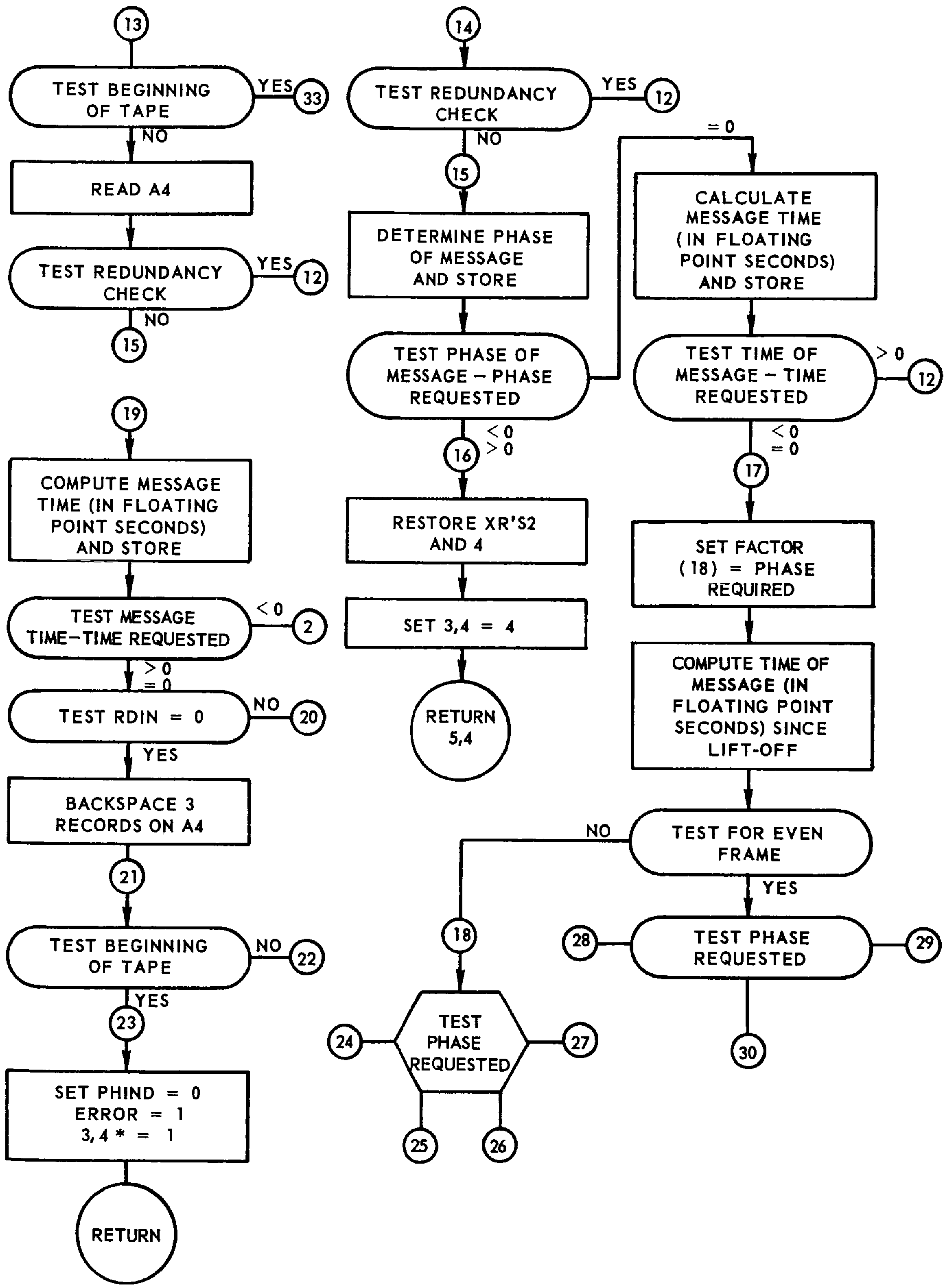


FIGURE 4-12. HIGH SPEED OUTPUT PROCESSOR PROGRAM (DONOUT) *(Sheet 4 of 10)*

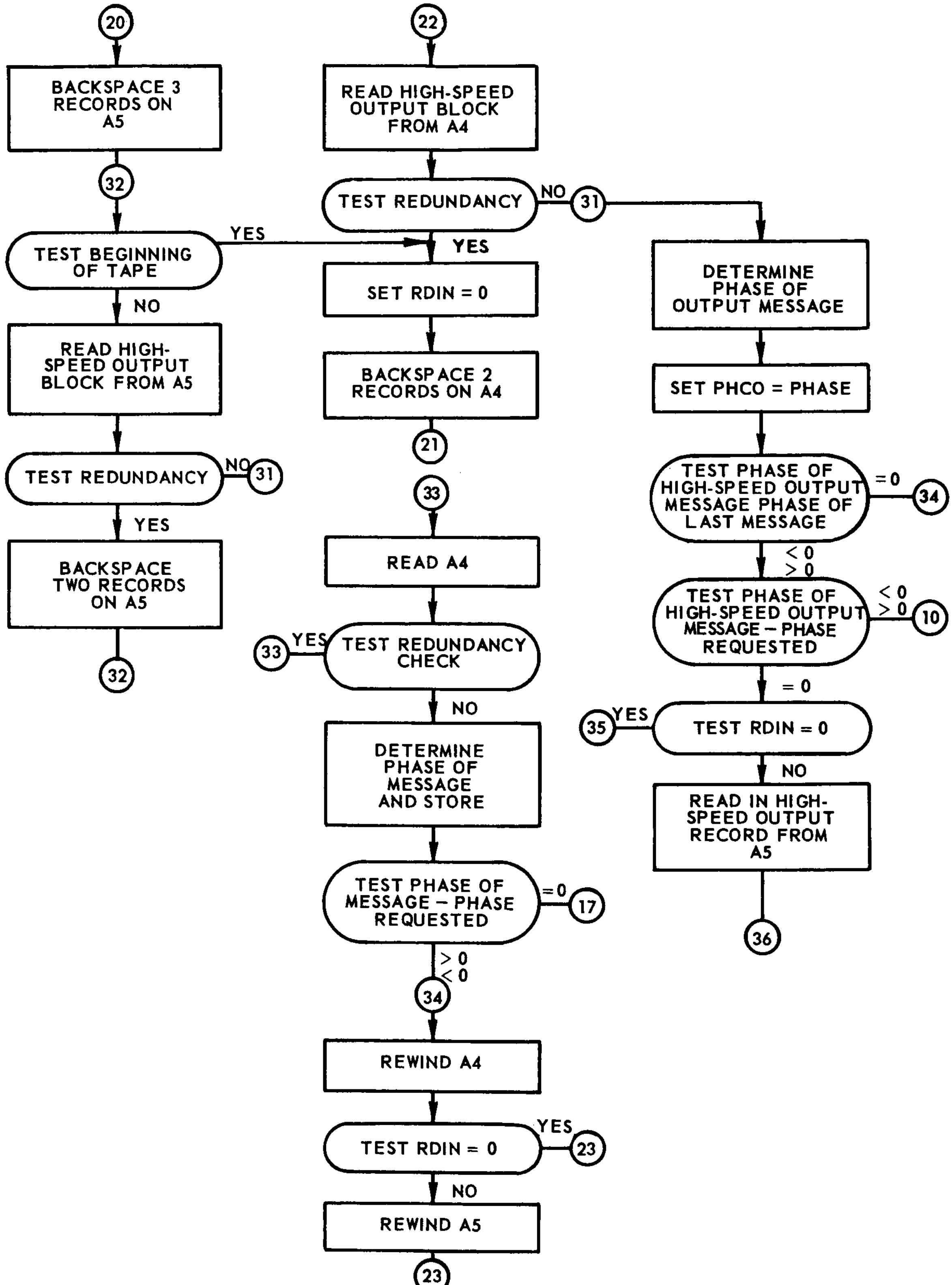


FIGURE 4-12. HIGH SPEED OUTPUT PROCESSOR PROGRAM (DONOUT) *(Sheet 5 of 10)*

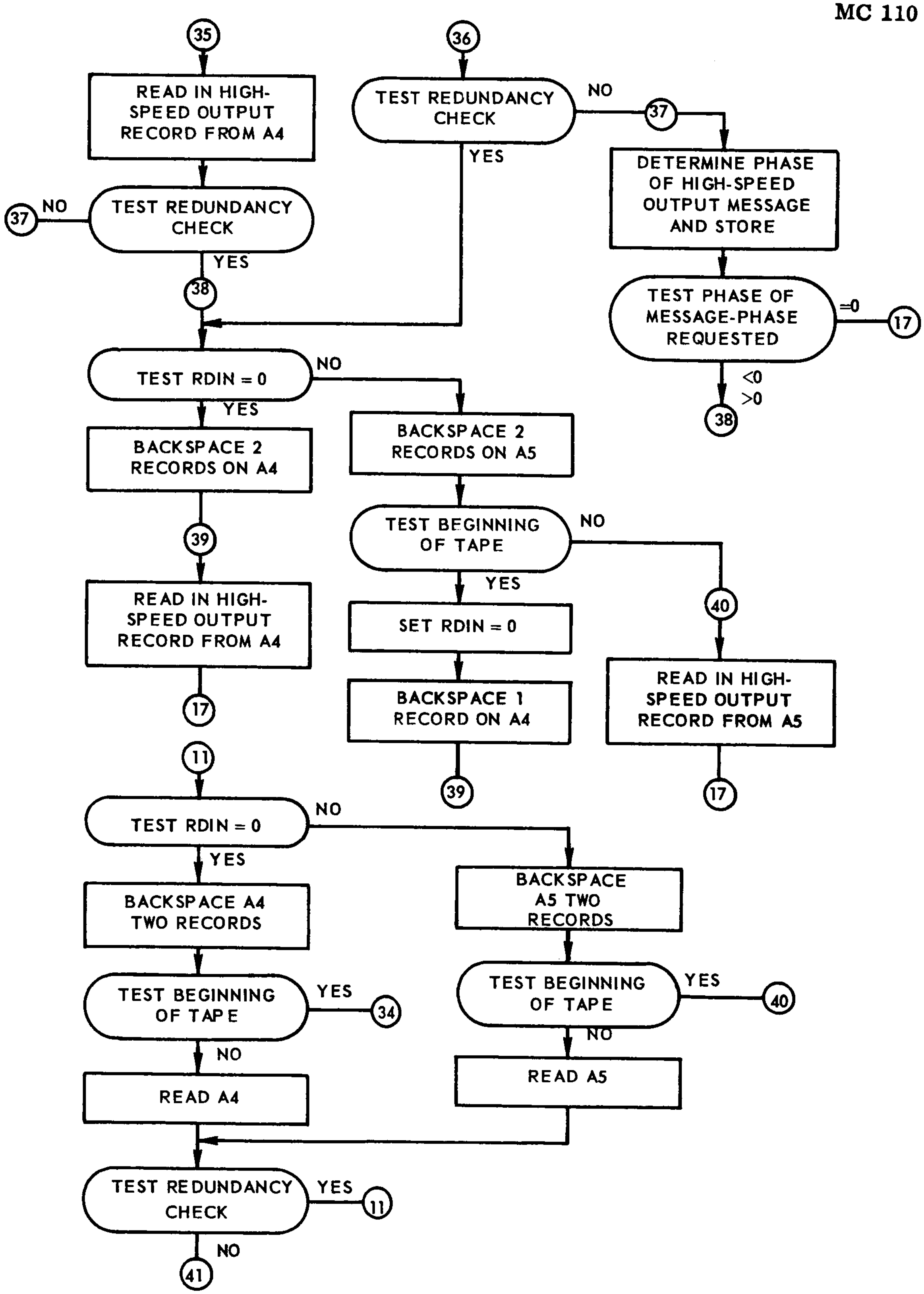


FIGURE 4-12. HIGH SPEED OUTPUT PROCESSOR PROGRAM (DONOUT) *(Sheet 6 of 10)*

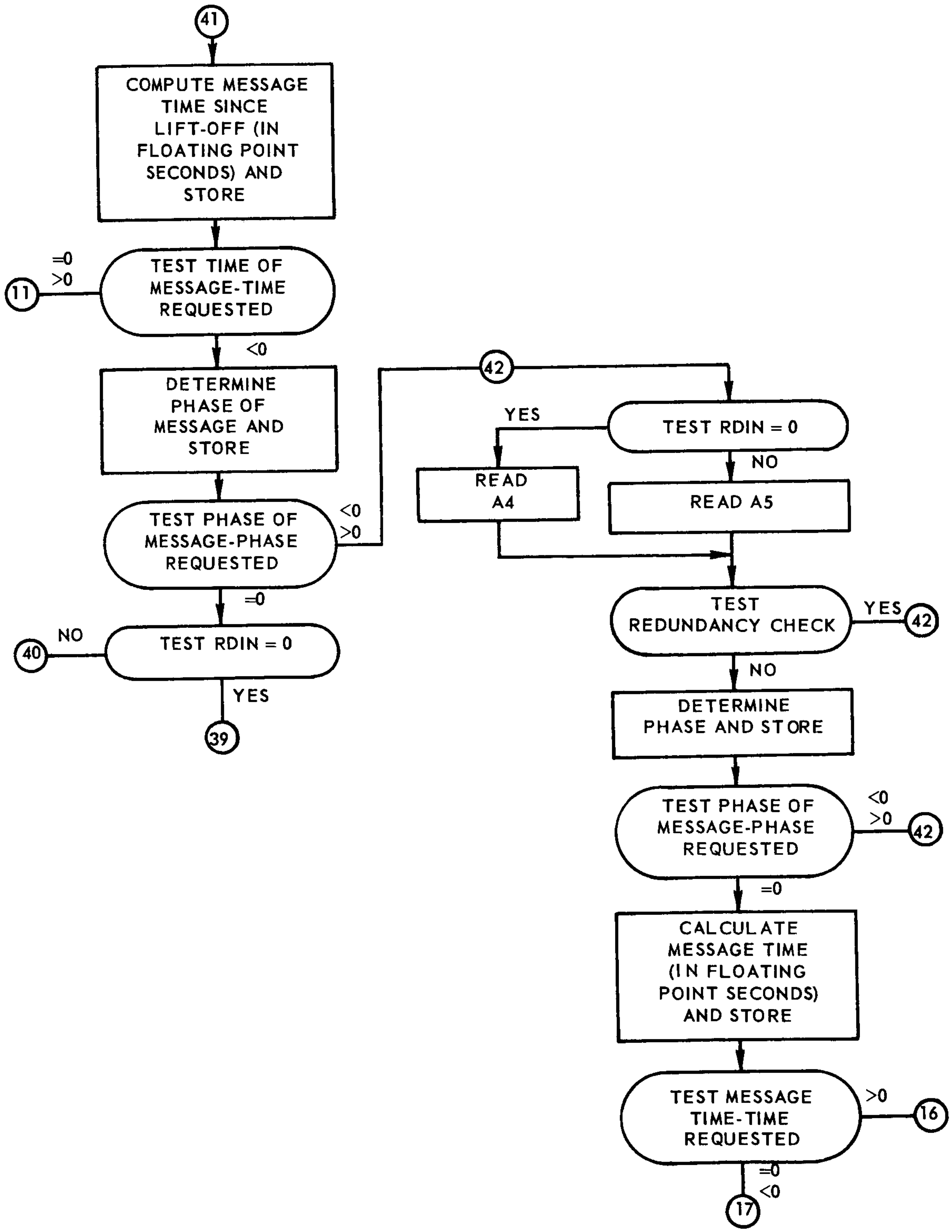


FIGURE 4-12. HIGH SPEED OUTPUT PROCESSOR PROGRAM (DONOUT) *(Sheet 7 of 10)*

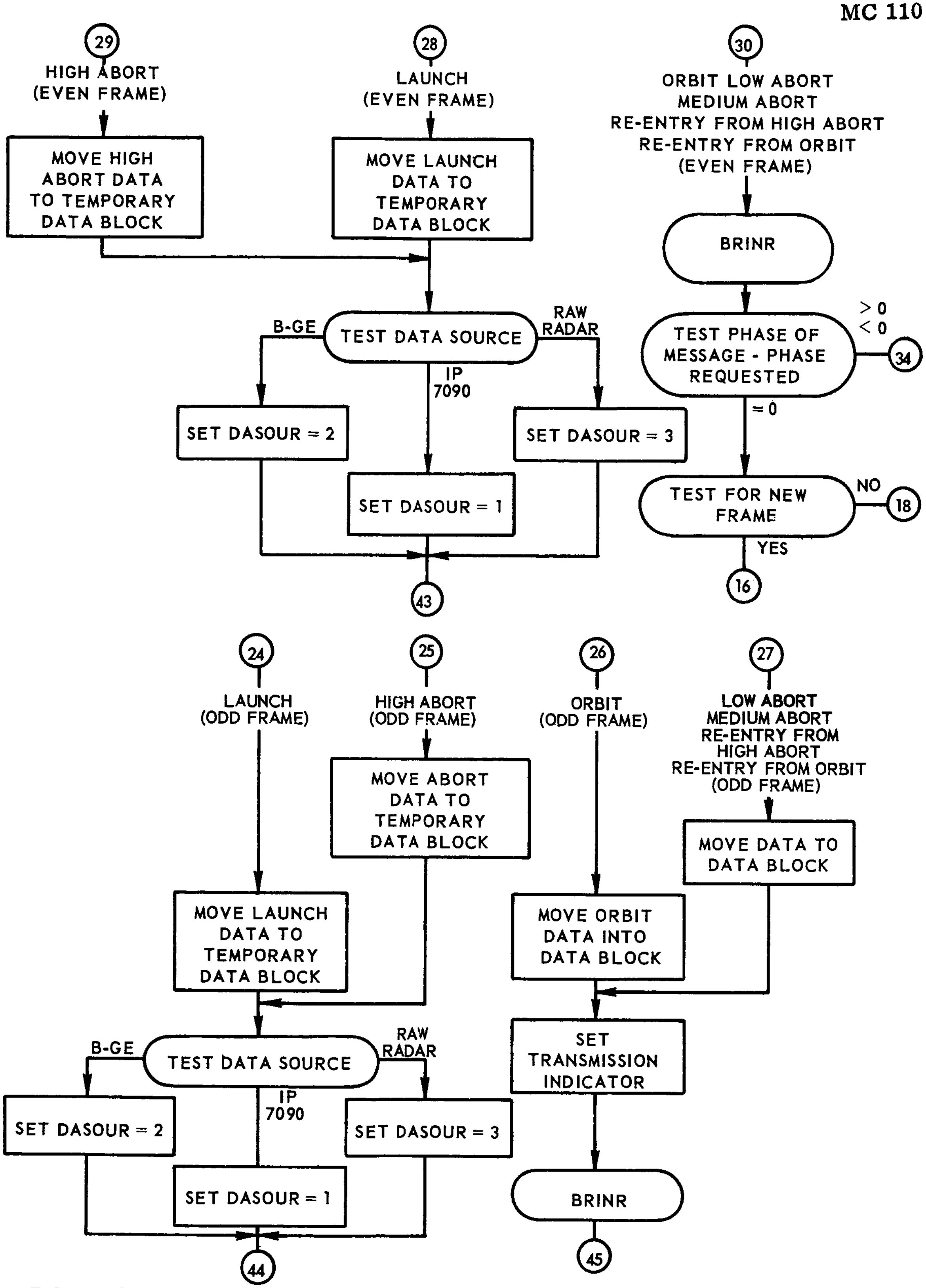


FIGURE 4-12. HIGH SPEED OUTPUT PROCESSOR PROGRAM (DONOUT) *(Sheet 8 of 10)*

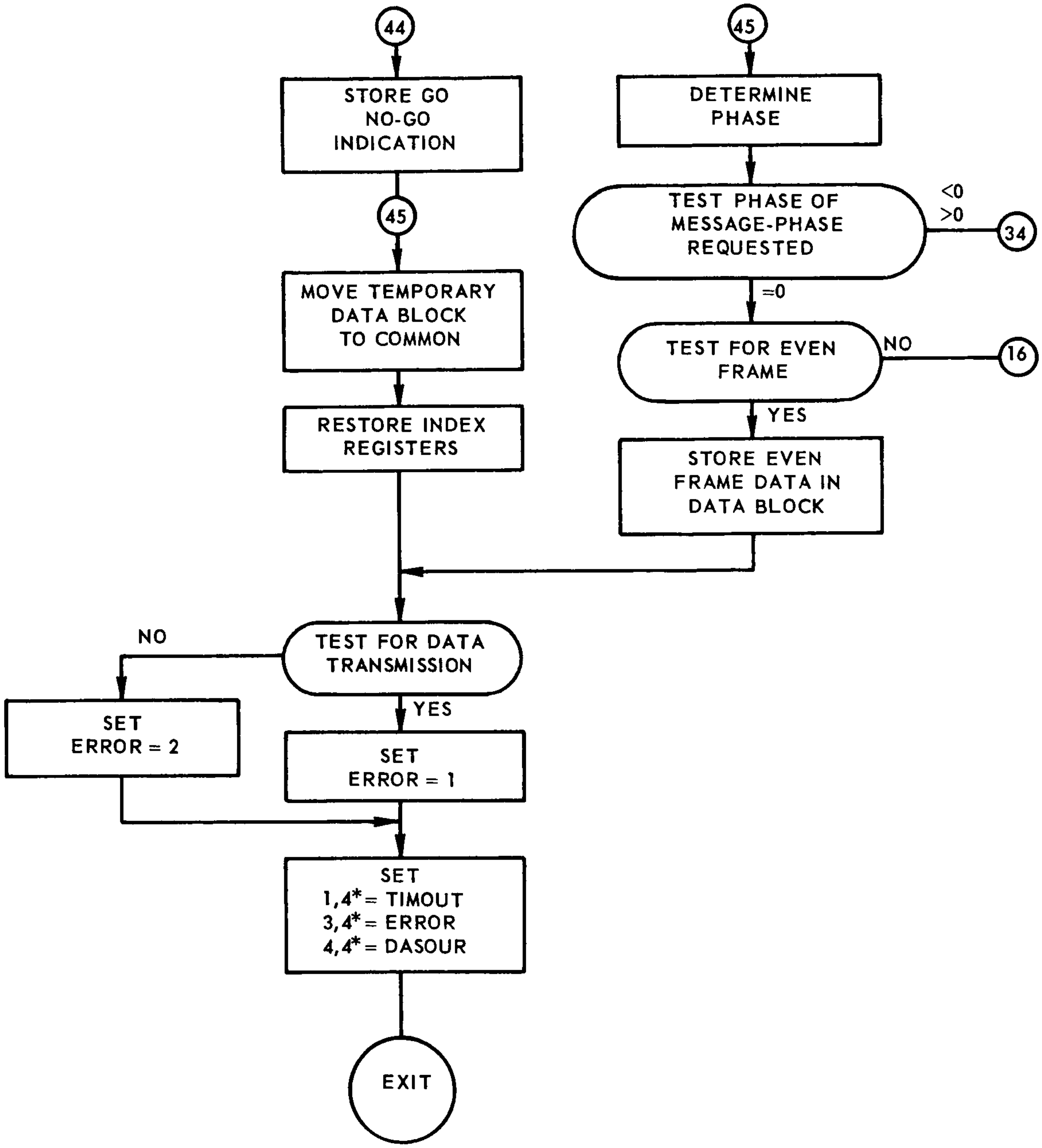


FIGURE 4-12. HIGH SPEED OUTPUT PROCESSOR PROGRAM (DONOUT) *(Sheet 9 of 10)*

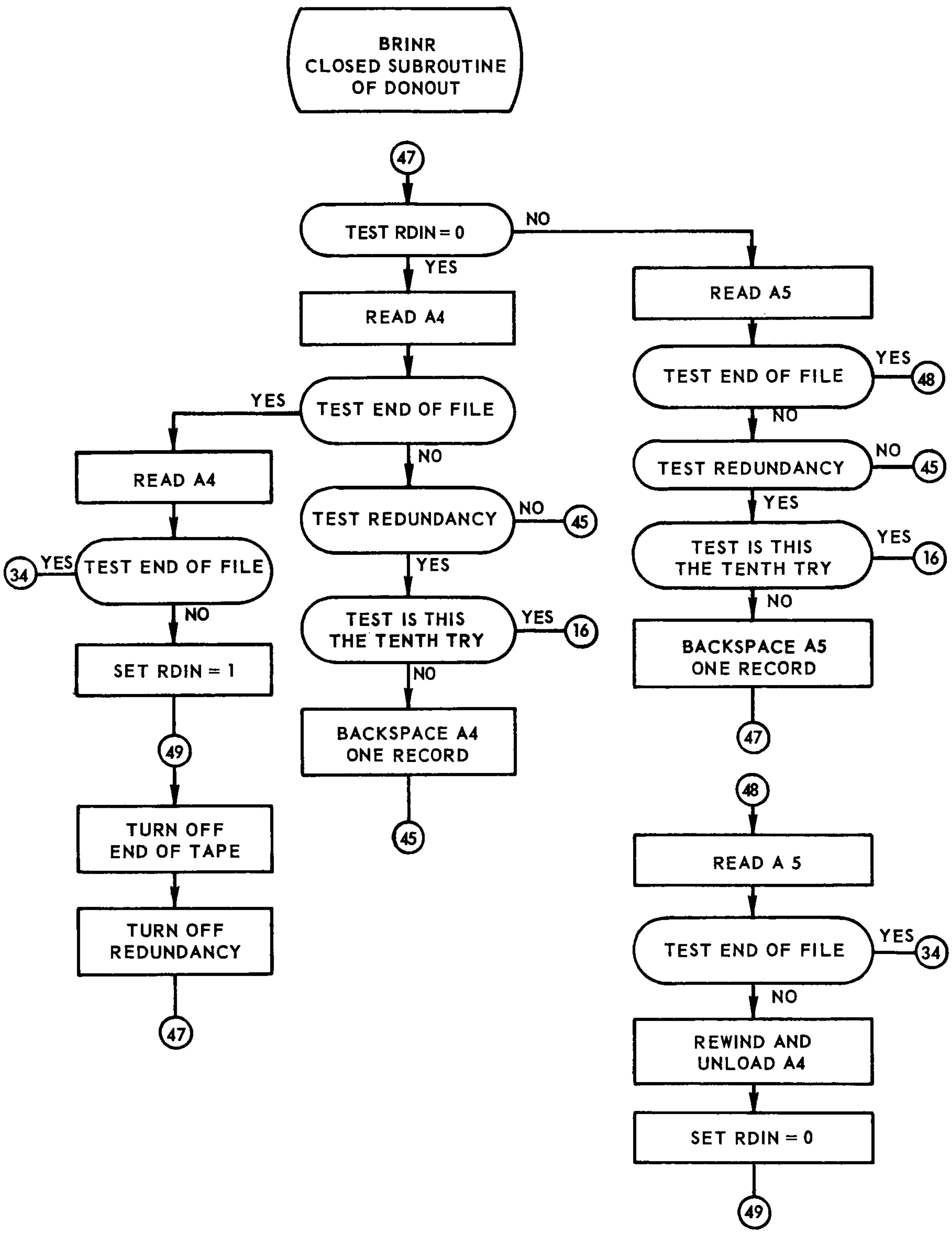


FIGURE 4-12. HIGH SPEED OUTPUT PROCESSOR PROGRAM (DONOUT) *(Sheet 10 of 10)*

4.4 HIGH-SPEED INPUT PROCESSOR PROGRAM (DONIN)

This program processes the high-speed input tape (B4) created by SORTER with data from the Mercury Program System log tape(s). The processed data, position and velocity vectors with an associated vector time, are placed into COMMON storage for later use in any mission situation that is characterized by associated high-speed input. The flow chart for DONIN is shown in Figure 4-13.

4.4.1 Input Requirements

Two parameters in the calling sequence of DONIN specify the vector time and data source of a record on the high-speed input tape. In addition to these input parameters (see Subsection 4.4.4), the B4 tape written by SORTER is input to DONIN. This program assumes 20-word records on this tape with the following format:

Word 1	Data Source (an octal integer), i.e.,	
	000 001 000 000	IP 7090 Computer
	000 002 000 000	B-GE Guidance Computer
	000 003 000 000	Raw radar
Words 2-11	Not used	
Word 12	Vector time (floating point seconds)	
Words 13-15	Position vectors (floating point feet)	
Words 16-18	Velocity vectors (floating point feet/second)	
Words 19-20	Not used	

4.4.2 Output Requirements

DONIN outputs seven quantities, three position vectors and three velocity vectors with the associated vector time. These are read into COMMON storage locations (see Subsection 4.4.4) indicated by the data source.

4.4.3 Method

The high-speed input tape is searched until the data source requested in the call statement is matched by a data source in one of the B4 records. Then, if the time of the high-speed input message being inspected is less than the time requested, control is transferred back to read in the next record. If the time of the message is greater than the time requested, the program backspaces until a message of the correct time and phase is determined. If the beginning of the B4 tape has been reached and the time requested is not found, an error return exit is made from the program.

4.4.4 Usage

Call Statement

CALL DONIN	(TEMPO1, NDATA, NOYES)
TEMPO1	contains the requested vector time
NDATA	is the data source indicator:
	1 specifies IP 7090 data
	2 specifies B-GE data
	3 specifies raw radar data
NOYES	is a return indicator
	1 indicates an unsuccessful return; a time and data source corresponding to the requested time and data were not found on B4. Vectors and vector time were not transferred to COMMON storage.
	2 indicates a successful return; a time and data source corresponding to the requested time and data source were found on B4. Vectors and vector time were transferred to COMMON storage.

Storage

The COMMON storage locations used for the position and velocity vectors and vector time are as follows:

B-GE position and velocity vectors: XIP, YIP, ZIP, XIP1, YIP1, ZIP1.

IP 7090 position and velocity vectors: XI, ETA, ZETA, XI1, ETA1, ZETA1.

Vector time: T

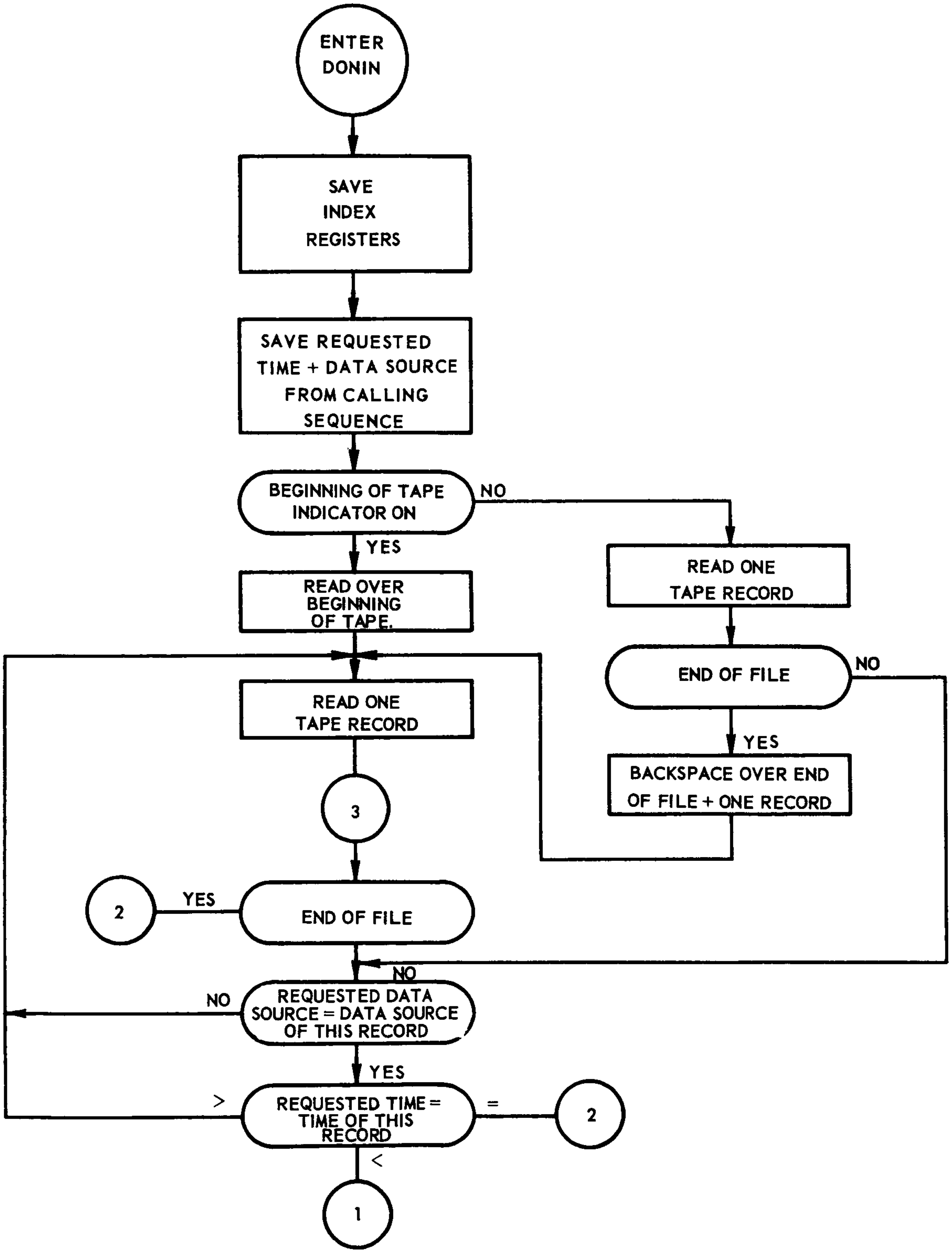


FIGURE 4-13. HIGH-SPEED INPUT PROCESSOR PROGRAM (DONIN) *(Sheet 1 of 2)*

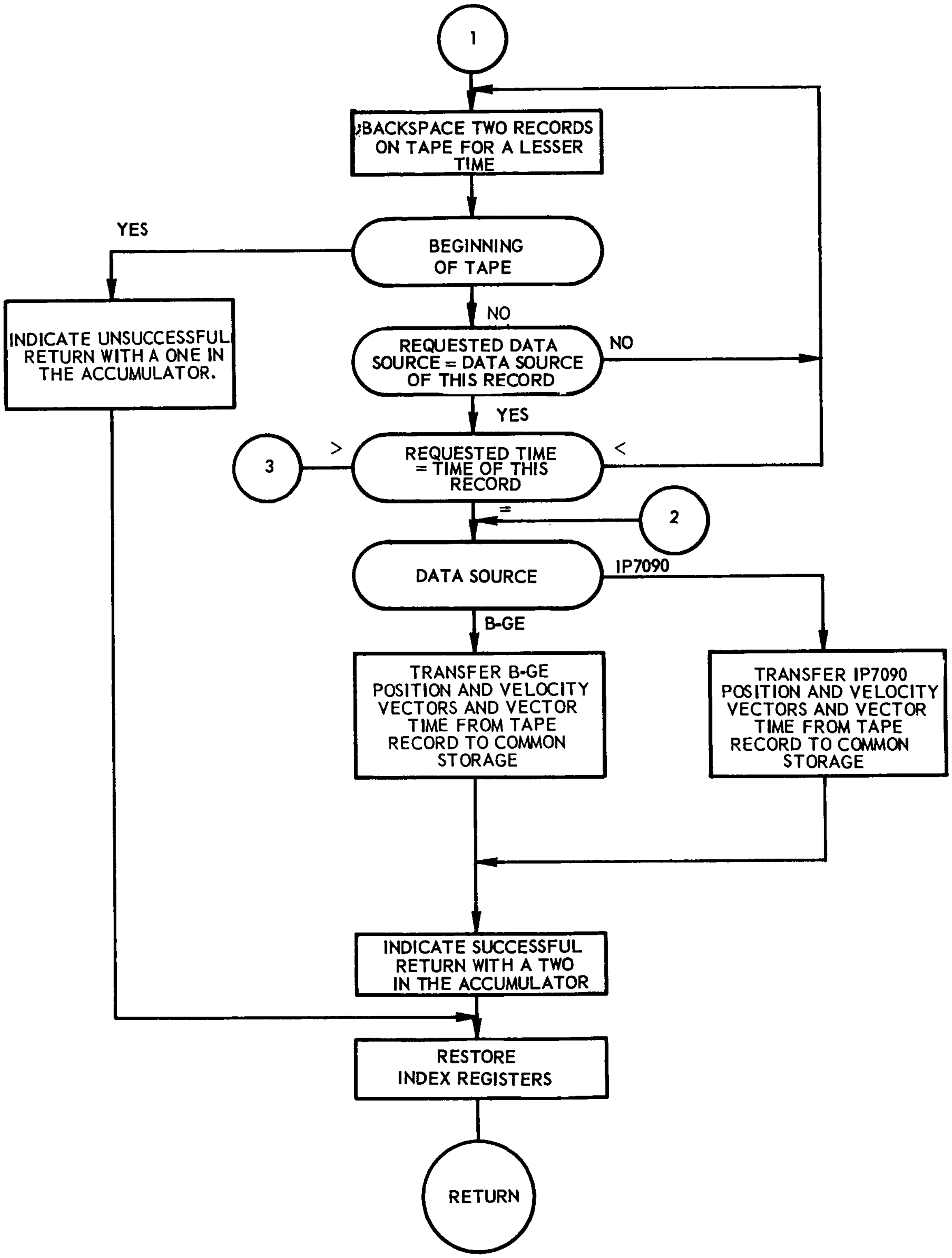


FIGURE 4-13. HIGH-SPEED INPUT PROCESSOR PROGRAM (DONIN) *(Sheet 2 of 2)*

SECTION 5
PHASE PROCESSOR PROGRAMS

Two types of phase processing programs are used in the Postflight Reporter Program. They are designated as the Major and Minor Processors. There are three Major Processor programs. They provide data for the launch, abort, orbit and re-entry phases. Monitor selects information from these data to write the paragraphs for each applicable phase in the postflight report.

The Minor Processor programs (nine in number) are subordinate to the Major phase processors. These Minor programs facilitate processing by performing particular important calculations for use specifically by the Major Processors.

5.1 MAJOR PROCESSOR PROGRAMS

There are three major phase processor programs, LAUNCH, ABORT and RENTER. All three produce data used in writing their respective paragraphs in the postflight report. Each paragraph represents an independent reapplication of phase processing, so that each set of data is treated out of context. The LAUNCH program is used whenever there is associated high-speed input data, and the launch, abort or reentry displays are driven. ORBIT is used when orbit displays are driven. RENTER is used whenever re-entry displays are driven or when abort displays are driven. No associated high-speed input is involved in the RENTER program.

5.1.1 Launch Phase Processor Program (LAUNCH)

The Launch Processor program produces launch and abort phase parameters for the postflight report. LAUNCH also produces reentry phase parameters. This program is used whenever high-speed input data are available. For the Mercury Atlas (MA) launch phase, this program uses high-speed processed data from either the Impact Predictor 7090 Computer or the Burroughs-General Electric Guidance Computer. LAUNCH also uses high-speed input data for Mercury Redstone (MR) launch and abort situations. Almost all of the processing for the MA abort phase is done by the program RENTER (see Subsection 5.1.3).

The flow chart for LAUNCH is shown in Fig. 5-1.

5.1.1.1 Input Requirements

All input parameters and constants for this program are placed into COMMON storage prior to its entry. These data are used in the equations that produce the output (equations are shown in Subsection 5.1.1.3). In addition, other processor programs are used by LAUNCH. These subordinate programs, which are required for launch phase processing are:

Atmospheric Density Processor Program (ATMOS)

B-GE Reference Frame Conversion Program (GECNV)

Stagnation Heat Rate Processor Program (HEAT)

IP 7090 Reference Frame Conversion Program (IPCNV)

Velocity Component Processor Program (JOHN)

True Inertial Coordinate Conversion Program (MERCNV)

Raw Radar Conversion Program (RAWCNV)*

Re-entry Phase Processor Program (RENTER)

Range From Launch Pad Processor Program (RFLP)

5.1.1.2 Output Requirements

The LAUNCH program outputs data for indicators in its call statement (see Subsection 5.1.1.4). The indicators show the phase displayed and the data source used. however, the program also outputs parameters that are placed into COMMON storage before control is returned to Monitor. The computation of these parameters involves the equations shown in Subsection 5.1.1.3. The LAUNCH output parameters are used to write the Launch phase paragraph of the Postflight Report. They are listed below in order of their appearance on the flow chart.

Radial distance of the capsule (r)

Height above spherical earth ($r - \bar{R}$)

Speed in the inertial frame (V_i)

Geocentric latitude (L_C)

*This program is not yet a part of the Postlight Reporter Program.

Geodetic latitude (L_D)

Height above oblate earth (h_e)

Flight path angle in inertial frame (γ_i)

Longitude of capsule (λ)

Heading angle in inertial frame (ψ_i)

Heading angle in rotational frame (ψ_e)

Speed in rotational frame (V_e)

Flight path in rotational frame (γ_e)

Mach number (M)

Reynold's number (R_N)

Stagnation beat rate (q_S)

Radial component of acceleration, g's (A_r)

5.1.1.3 Method

The equations used to calculate the outputs of LAUNCH are given below in the order of their appearance on the flow chart.

$$r = \sqrt{\underline{\bar{X}}^2 + \underline{\bar{Y}}^2 + \underline{\bar{Z}}^2}$$

$$r - \bar{R} = (r) - \bar{R}$$

$$V_i = \sqrt{\underline{\dot{X}}^2 + \underline{\dot{Y}}^2 + \underline{\dot{Z}}^2}$$

$$L_C = \tan^{-1} \frac{\underline{\bar{Z}}}{\sqrt{\underline{\bar{X}}^2 + \underline{\bar{Y}}^2}}$$

$$L_D = \tan^{-1}\left[\left(\frac{a_c}{b_c}\right)^2 \tan L_C\right]$$

$$e_2 = 1/(1-f)^2$$

$$x_\nu = \frac{a_e}{\sqrt{\cos^2 L_C + e_2 \sin^2 L_C}}$$

$$h_e = r - x_\nu$$

$$\Delta\lambda_i = \tan^{-1} (\underline{Y} / \underline{X})$$

$$\gamma_i = \sin^{-1} (\dot{x} / V_i)$$

$$\lambda = \Delta\lambda_i - \omega_e t - \nu_a$$

$$\psi_i = \tan^{-1} (\dot{y}_i / \dot{z})$$

$$\psi_e = \tan^{-1} (\dot{y} / \dot{z}), \text{ where } \dot{y} = y_i - \omega_e r \cos L_C$$

$$V_e = \sqrt{\dot{x}^2 + \dot{y}^2 + \dot{z}^2}$$

$$\gamma_e = \sin^{-1} (\dot{x} / V_e)$$

$$M = V_e / C_S$$

$$R_N = V_e / \nu$$

$$q_D = 1/2 \, \rho \, V_e^2$$

$$A_r = V_e^2 / r$$

$$r_1 = \frac{a_c}{\sqrt{\cos^2 L_C + (a_c/b_c)^2 \sin^2 L_C}}$$

5.1.1.4 Usage

Call Statement

CALL LAUNCH (J23, NDATA)

The parameter J23 is a phase indicator.

1 indicates launch phase (MA and MR)

2 indicates abort phase (MR)

5 indicates high or medium abort phase (MA)

6 indicates low abort phase (MA)

NDATA is a data source indicator.

1 indicates IP 7090 data

2 indicates B-GE data

3 indicates raw radar data

5.1.2 Orbit Phase Processor Program (ORBIT)

ORBIT produces orbit phase parameters for the postflight report. The program is used whenever orbit displays are driven (there is no high-speed input). The flow chart for the ORBIT program is shown in Figure 5-2.

5.1.2.1 Input Requirements

All input parameters and constants used in the ORBIT output equations are placed into COMMON storage before control is transferred to the main program. ORBIT also uses the subroutines JIM (Position and Velocity Processor Program) and ODOT ($\dot{\omega}$ and $\dot{\Omega}$ Processor Program).

5.1.2.2 Output Requirements

ORBIT produces certain orbit phase parameters and places them into COMMON storage prior to its exit. These ORBIT parameters are used to write the orbit phase of the postflight report. They are listed below in order of their appearance on the flow chart.

Radial distance to the capsule (r)

Semi-major axis (a)

Geocentric latitude (L_C)

Height of apogee above the spherical earth of the apogee point (h_a)

Components of the position vector $(\vec{r})$ for $\underline{X}$, $\underline{Y}$, $\underline{Z}$ in inertial reference frame

Semi-latus rectum (p)

Argument of perigee . . . for proper quadrant (ω)

Sidereal period (T)

True anomaly (θ_1)

Eccentric anomaly (E)

Mean anomaly (M)

Flight path angle in the inertial frame (γ_i)

Heading angle in the inertial frame (ψ_i)

Height above the oblate earth (h_e)

Velocity in the rotational system (V_e)

Flight path angle in the rotational system (γ_e)

Heading angle in the rotational system (ψ_e)

Time to perigee passage (t_p)

5.1.2.3 Method

In order of their appearance on the flow chart, the equations required to calculate orbital output parameters are given below.

$$r = (r - \overline{R}) + (\overline{R})$$

$$a = (a - \overline{R}) + (\overline{R})$$

$$L_C = \tan^{-1}\left[\left(\frac{b_c}{a_c}\right)^2 \tan L_D\right]$$

$$h_a = a(1 + e) - \overline{R}$$

$$r = (\underline{X}, \underline{Y}, \underline{Z}): \begin{pmatrix} \underline{X} \\ \underline{Y} \\ \underline{Z} \end{pmatrix} \begin{pmatrix} r \cos L_C \cos \lambda \\ r \cos L_C \sin \lambda \\ r \sin L_C \end{pmatrix}$$

$p = q\ (1+e)$ where $q = a\ (1-e)$

$$\sin\omega = \frac{p \cos i \cos \Omega - q \sin \Omega}{\Delta}$$

$$\cos\omega = \frac{q \cos i \cos \Omega + p \sin \Omega}{\Delta}$$

$$\omega = \tan^{-1}\left(\frac{\sin\omega}{\cos\omega}\right), \text{ where } \Delta \equiv \cos^2 i \cos^2 \Omega + \sin^2 \Omega$$

$$\vec{P} = (P_x, P_y, P_z), \text{ where } \begin{pmatrix} P_x \\ P_y \\ P_z \end{pmatrix} = \begin{pmatrix} \sin\omega \cos i \sin\Omega + \cos\omega \cos\Omega \\ p \\ \sin i \sin\omega \end{pmatrix}$$

$$\vec{Q} = (Q_x, Q_y, Q_z), \text{ where } \begin{pmatrix} Q_x \\ Q_y \\ Q_x \end{pmatrix} = \begin{pmatrix} -\cos\omega \cos i \sin\Omega - \sin\omega \cos\Omega \\ q \\ \sin i \cos\omega \end{pmatrix}$$

$$\theta_1 = \tan^{-1}\left(\frac{\sin\theta_1}{\cos\theta_1}\right), \text{ where } \sin\theta_1 = \frac{\vec{r} \cdot \vec{Q}}{r} \text{ and}$$

$$\cos\theta_1 = \frac{a}{r}(\cos E - e)$$

$$E = \tan^{-1}\left(\frac{\sin E}{\cos E}\right), \text{ where } \sin E = \frac{r}{p}\sin\theta\sqrt{1-e^2} \text{ and}$$

$$\cos E = \frac{e \cos E}{e} \text{ where } e\cos E = 1 - r/a$$

$$V_i = (\dot{\underline{X}}, \dot{\underline{Y}}, \dot{\underline{Z}}), \text{ where } \vec{V}_i = \frac{\vec{Q}(\cos\theta_1 + e) - \vec{P}\sin\theta_1}{\sqrt{p}}$$

$$M = E - e \sin E$$

$$\gamma_i = \sin^{-1}\left(\frac{\vec{r}\ \vec{V}_i}{r\ V_i}\right)$$

$$\psi_i = \sin^{-1} \left(\frac{\cos i}{\cos L_C} \right)$$

$$h_e = r - \frac{a_e}{\sqrt{\cos^2 L_C + e_2 \sin^2 L_C}}$$

$$\vec{V}_e = (V_x, V_y, V_z), \text{ where } \vec{V}_e = \vec{V}_i - \vec{\omega}_e \times \vec{r} \text{ and } \vec{\omega}_e = (0, 0, \omega_e)$$

$$\gamma_e = \sin^{-1} \left(\frac{V_i}{V_e} \sin \gamma_i \right)$$

$$\psi_e = \tan^{-1} \left(\frac{V_y}{V_z} \right)$$

$$t_p = M_a \sqrt{\frac{a}{\mu_e}}$$

5.1.2.4 Usage

Call statement

CALL ORBIT

5.1.3 Re-entry Phase Processor Program (RENTER)

RENTER produces re-entry phase parameters for the postflight report. This program is used whenever re-entry displays are driven (there is no associated high-speed input). It is also used when abort displays are driven and there is no associated high-speed input. The flow chart for the RENTER program is shown in Fig. 5-3.

5.1.3.1 Input Requirements

All input parameters and constants used in the RENTER equations are placed into COMMON storage prior to entry of the program. These equations produce output for RENTER. This program also uses other processor programs, which are:

Launch Phase Processor Program (LAUNCH)

Atmospheric Density Processor Program (ATMOS)

Position and Velocity Processor Program (JIM)

Velocity Component Processor Program (JOHN)

5.1.3.2 Output Requirements

RENTER outputs data for a phase indicator in its call statement (see Subsection 5.1.3.4). The program also outputs re-entry parameters that are placed into COMMON storage before its exit. The computation of these parameters involves the equations shown in Subsection 5.1.3.3. The RENTER output parameters are used to write the re-entry phase and sometimes the abort phase paragraph of the Postflight Report. They are listed below in the order of their appearance on the flow chart.

Radial distance of the capsule (r)

Geocentric latitude (L_C)

Heading angle in the inertial frame (ψ_i)

Height above the oblate earth (h_e)

Heading angle in the rotational system (ψ_e)

Speed in the rotational system (V_e)

Flight path angle in the rotational system (γ_e)

Mach number (M)

Reynold's number (R_N)

Dynamic pressure (q_D)

5.1.3.3 Method

The equations used to calculate the output parameters for the RENTER are shown below, in order of their appearance on the flow chart.

$$r = (r - \overline{R}) + \overline{R}$$

$$L_C = \tan^{-1} \left[\left(\frac{b_c}{a_c} \right)^2 \tan L_D \right]$$

$$\psi_i = \sin^{-1} \left(\frac{\cos i}{\cos L_C} \right)$$

$$\Delta\lambda_i = \tan^{-1} (\underline{Y}/\underline{X})$$

$$h_e = r - x_\nu \text{ , where } x_\nu = \frac{a_e}{\sqrt{\cos^2 L_C + e_2 \sin^2 L_C}}$$

$$\psi_e = \tan^{-1} (\dot{y}/\dot{z}), \text{ where } \dot{y} = \dot{y}_i - \omega_e \, r \cos L_C$$

$$V_e = \sqrt{\dot{x}^2 + \dot{y}^2 + \dot{z}^2}$$

$$\gamma_e = \sin^{-1} \left(\frac{\dot{x}}{V_e} \right)$$

$$M = V_e / C_S$$

$$R_N = V_e / \nu$$

$$q_D = 1/2 \, \rho \, V_e^2$$

$$r_1 = \frac{a_c}{\sqrt{\cos^2 L_C + \left(\frac{a_c}{b_c} \right)^2 \sin^2 L_C}}$$

5.1.3.4 Usage

Call Statement

CALL RENTER (J23)

J23 is a phase indicator

4 indicates re-entry phase

5 indicates high abort phase (MA)

6 indicates low abort phase (MA)

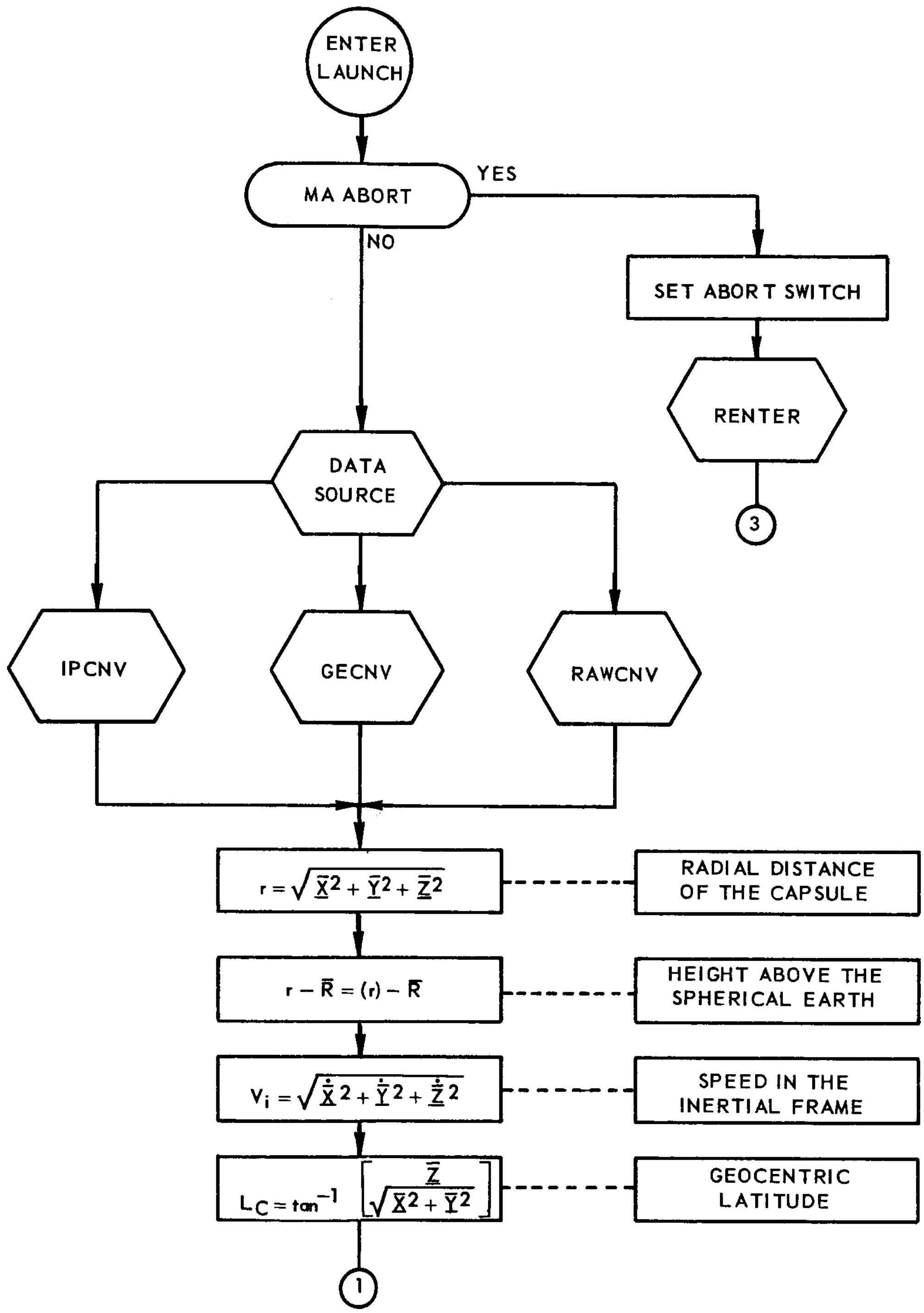


FIGURE 5-1. LAUNCH PHASE PROCESSOR PROGRAM (LAUNCH) *(Sheet 1 of 4)*

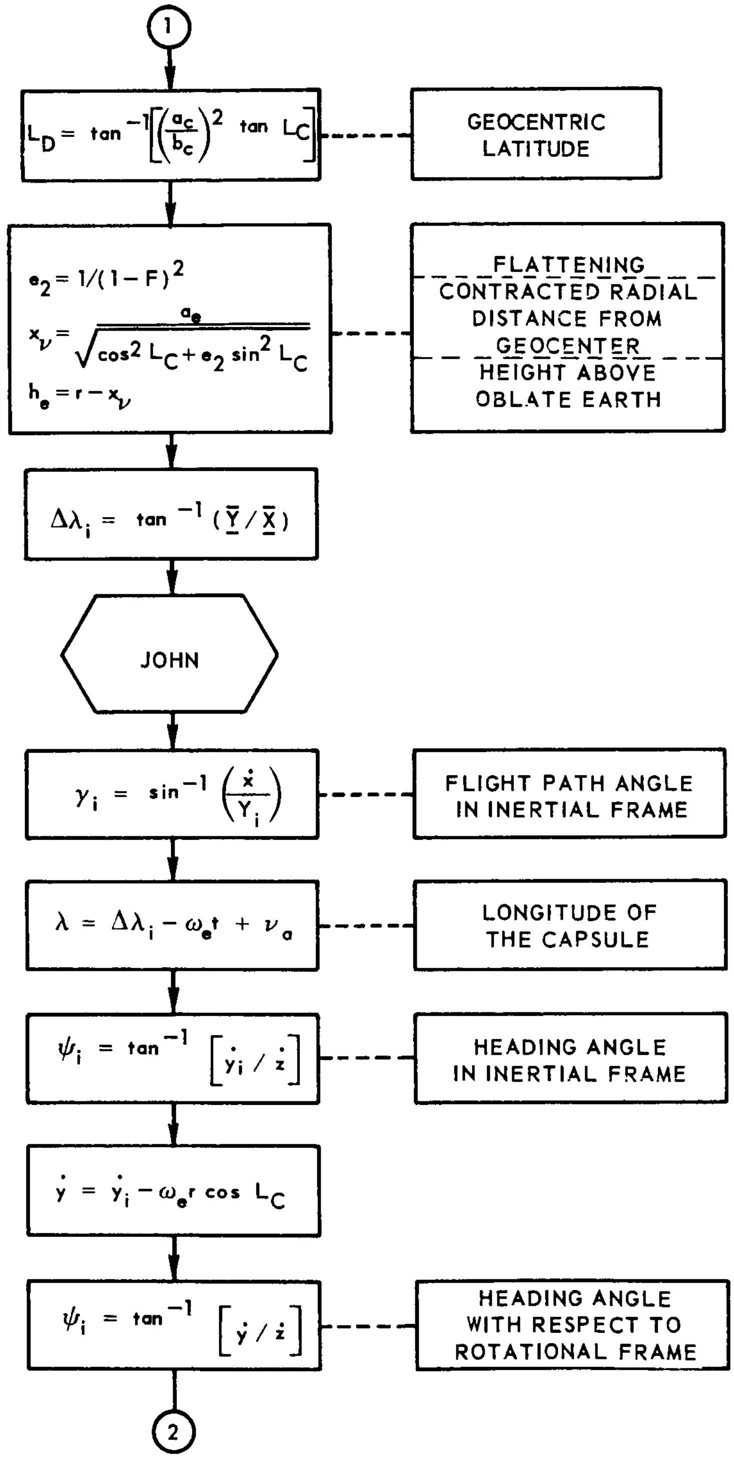


FIGURE 5-1. LAUNCH PHASE PROCESSOR PROGRAM (LAUNCH) *(Sheet 2 of 4)*

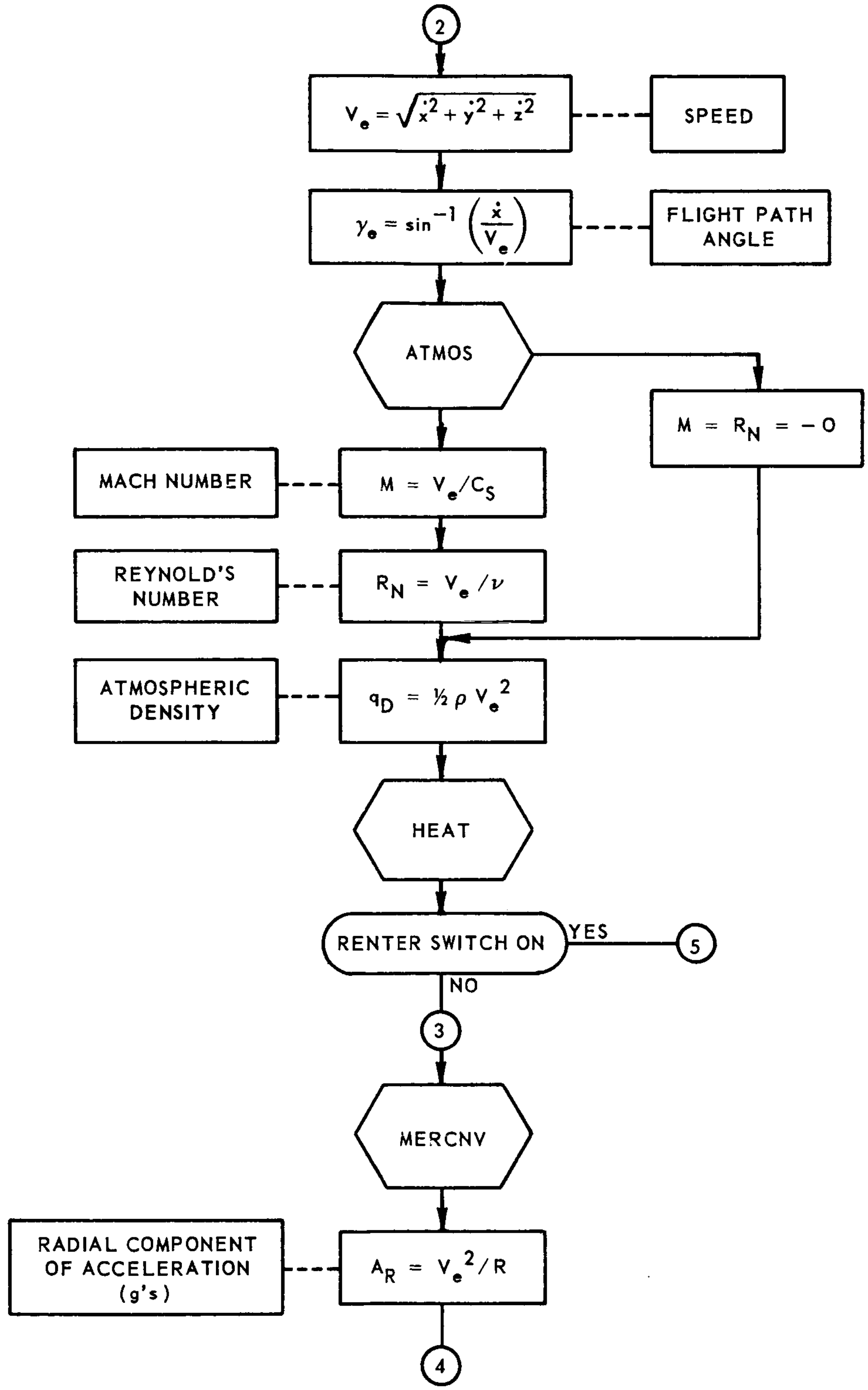


FIGURE 5-1. LAUNCH PHASE PROCESSOR PROGRAM (LAUNCH) *(Sheet 3 of 4)*

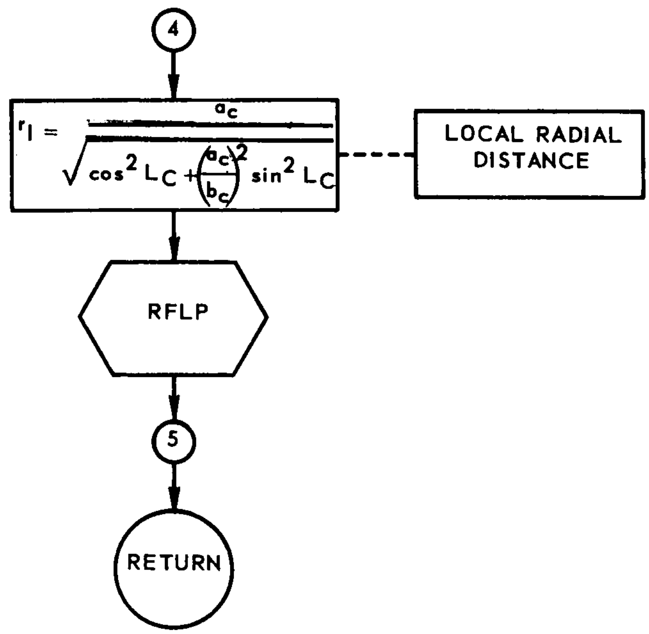


FIGURE 5-1. LAUNCH PHASE PROCESSOR PROGRAM (LAUNCH) *(Sheet 4 of 4)*

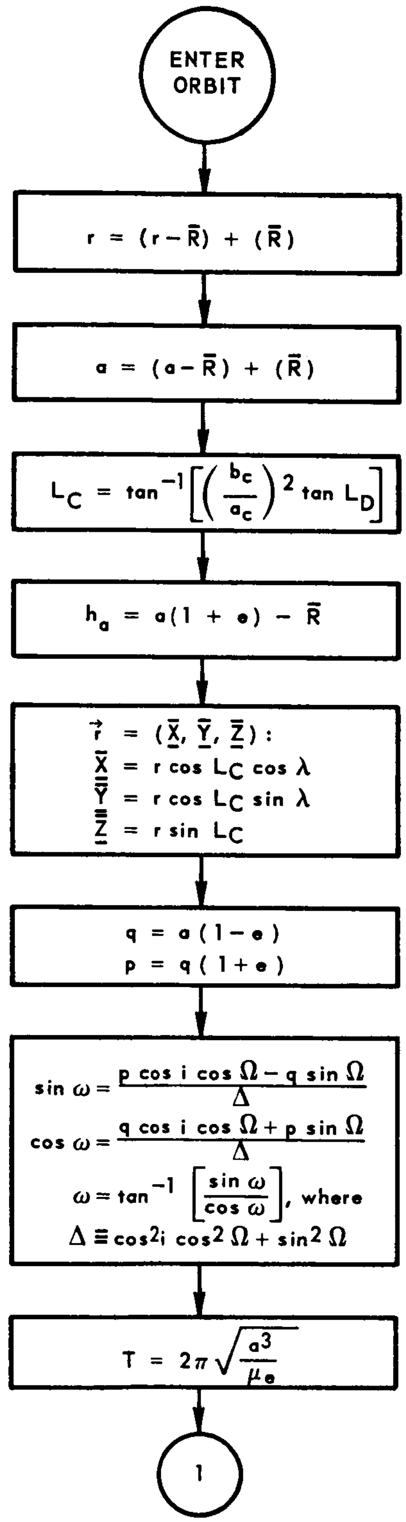


FIGURE 5-2. ORBIT PHASE PROCESSOR PROGRAM (ORBIT) *(Sheet 1 of 3)*

1

$$\vec{P} = (P_x, P_y, P_z):$$
$$P_x = \sin\omega \cos i \sin\Omega + \cos\omega \cos\Omega$$
$$P_y = P$$
$$P_z = \sin i \sin\omega$$

$$\vec{Q} = (Q_x, Q_y, Q_z):$$
$$Q_x = -\cos\omega \cos i \sin\Omega - \sin\omega \cos\Omega$$
$$Q_y = q$$
$$Q_z = \sin i \cos\omega$$

$$e \cos E = 1 - r/a$$
$$\cos E = \frac{(e \cos E)}{e}$$

$$\sin \Theta_1 = \frac{\vec{r} \cdot \vec{Q}}{r}$$
$$\cos \Theta_1 = \frac{a}{r}(\cos E - e)$$
$$\Theta_1 = \tan^{-1}\left[\frac{\sin\Theta_1}{\cos\Theta_1}\right]$$

$$\sin E = \frac{r}{p} \sin \Theta_1 \sqrt{1 - e^2}$$
$$E = \tan^{-1}\left[\frac{\sin E}{\cos E}\right]$$

$$\vec{V}_i = (\dot{\underline{X}}, \dot{\underline{Y}}, \dot{\underline{Z}}):$$
$$\vec{V}_i = \frac{\vec{Q}(\cos\Theta_1 + e) - \vec{P}\sin\Theta_1}{\sqrt{P}}$$

2

FIGURE 5-2. ORBIT PHASE PROCESSOR PROGRAM (ORBIT) *(Sheet 2 of 3)*

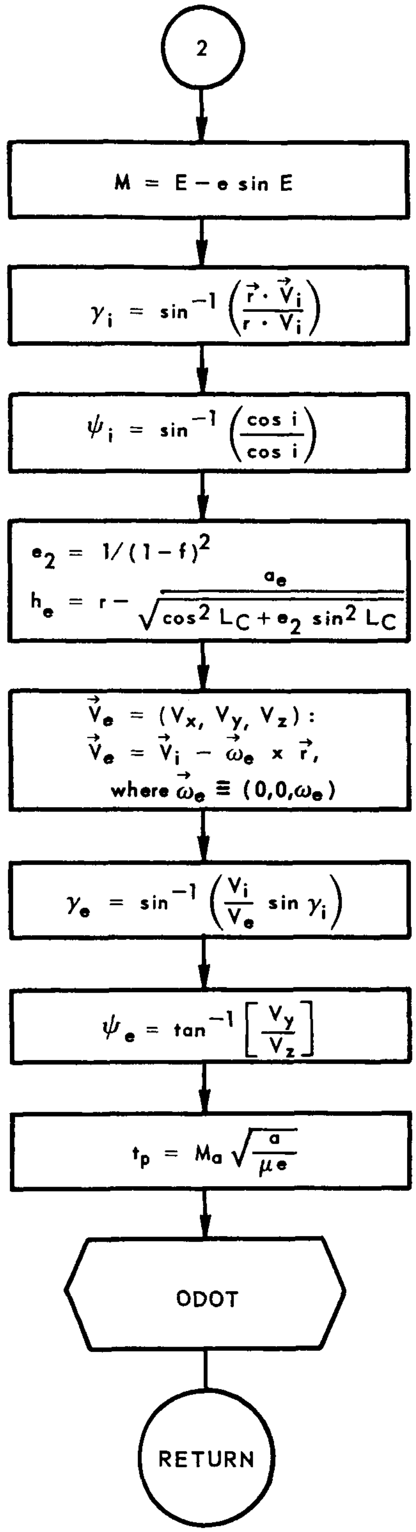


FIGURE 5-2. ORBIT PHASE PROCESSOR PROGRAM (ORBIT) *(Sheet 3 of 3)*

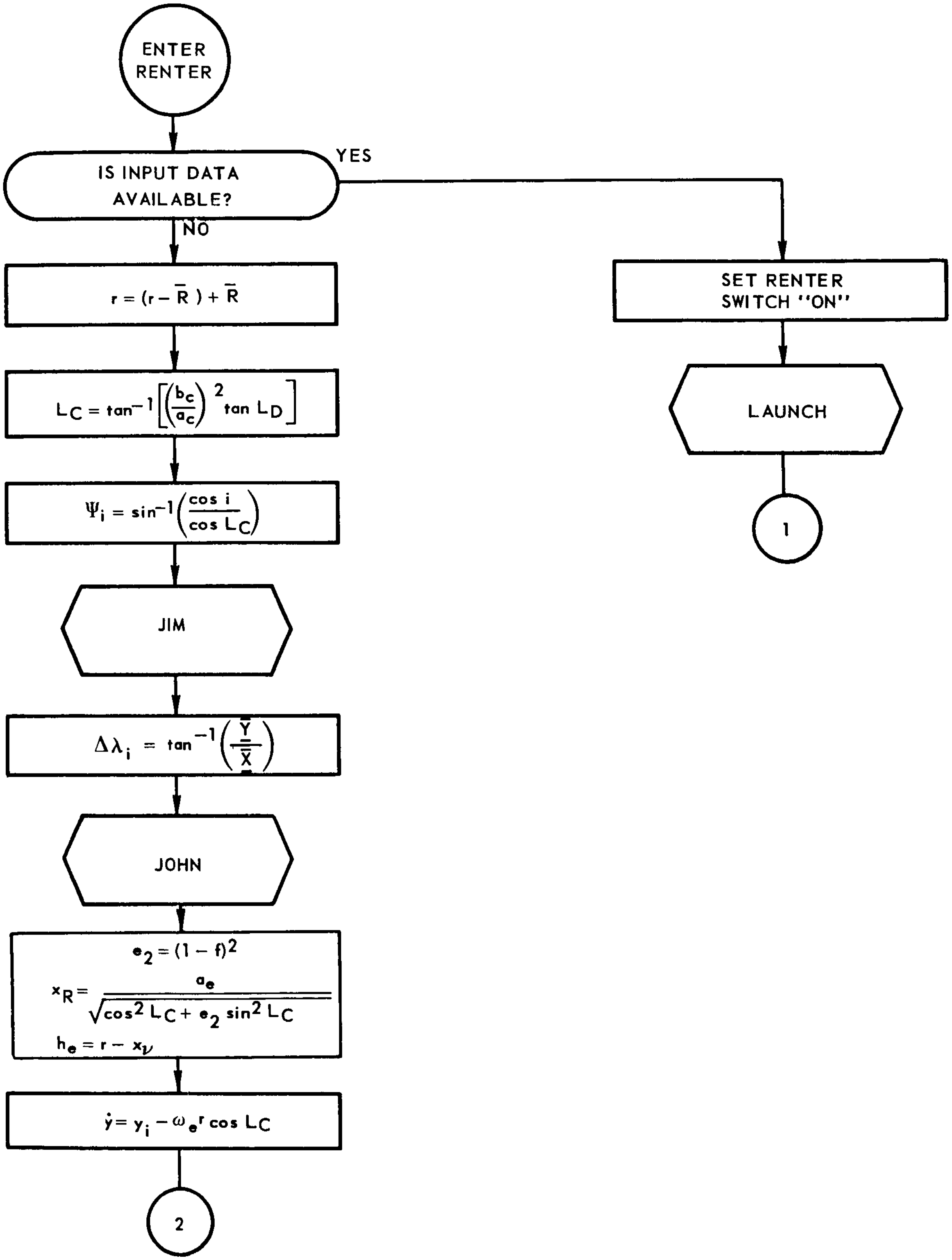


FIGURE 5-3. RE-ENTRY PHASE PROCESSOR PROGRAM (RENTER) *(Sheet 1 of 3)*

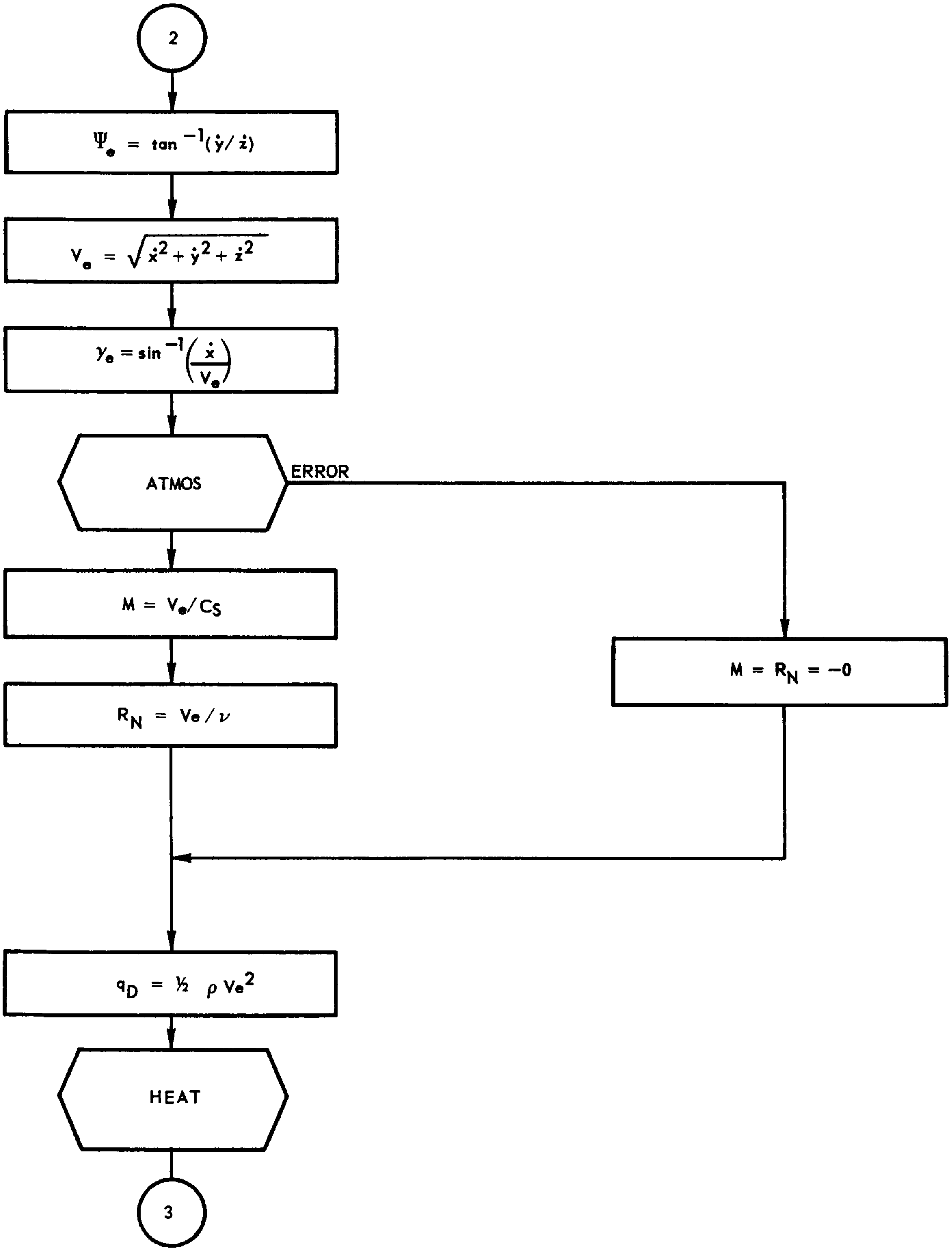


FIGURE 5-3. RE-ENTRY PHASE PROCESSOR PROGRAM (RENTER) *(Sheet 2 of 3)*

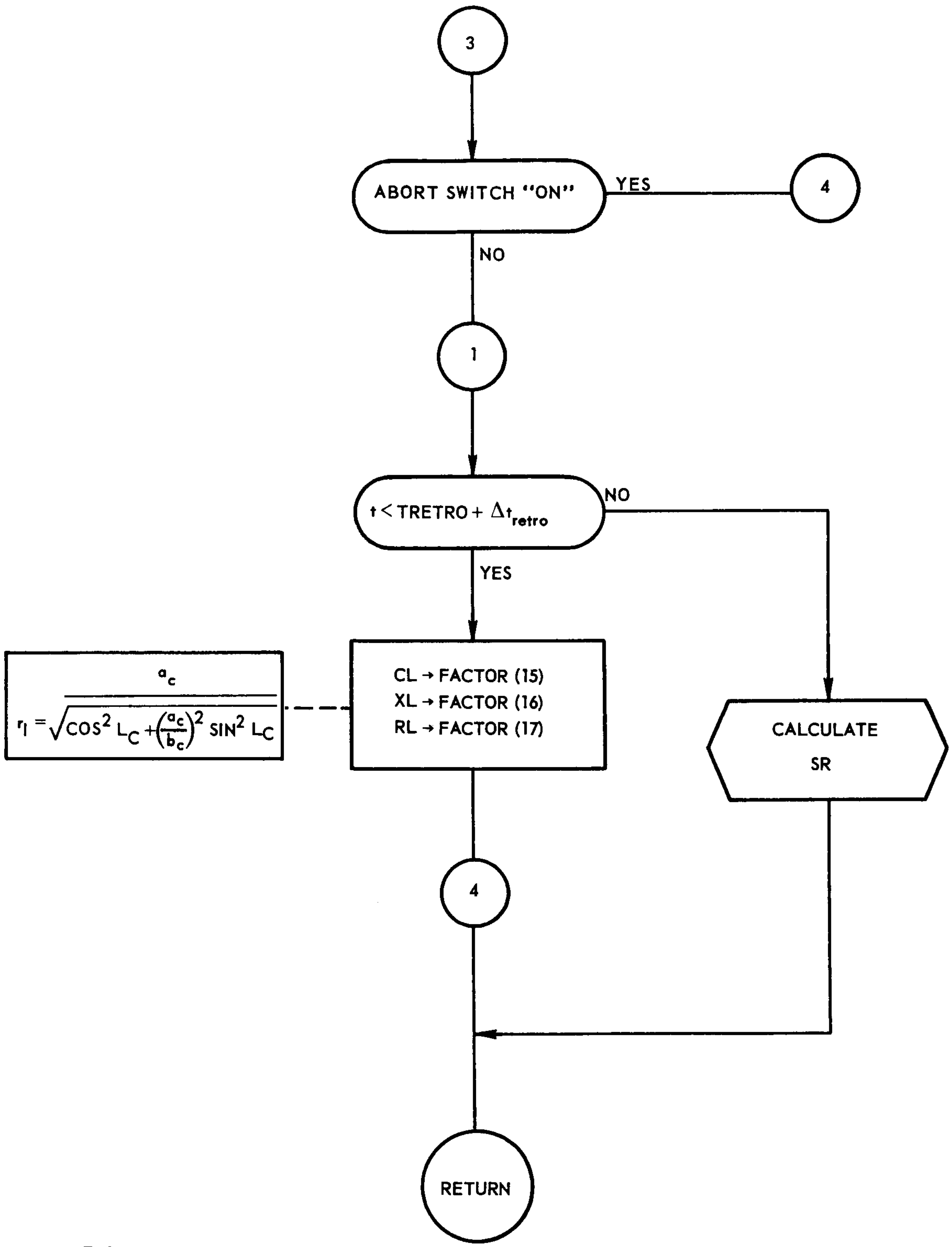


FIGURE 5-3. RE-ENTRY PHASE PROCESSOR PROGRAM (RENTER) *(Sheet 3 of 3)*

5.2 MINOR PROCESSOR PROGRAMS

The nine minor processor programs perform calculations for the major phase processors and are used only with these programs. There are two groups of minor processors. One group, consisting of IPCNV, GECNV, MERCNV, JIM, and JOHN, is concerned with coordinate conversion. The other group, consisting of ATMOS, HEAT, RFLP, and ODOT performs auxiliary calculations for the phase processors.

IPCNV, a subroutine of LAUNCH, converts IP 7090 reference frame $\vec{r}$ and $\vec{v}$ components to true inertial coordinates. GECNV, another LAUNCH subroutine, converts B-GE reference frame processed data ($\vec{r}$ and $\vec{v}$ components) to true inertial coordinates. MERCNV converts data from the true inertial frame to pad rectangular coordinates for LAUNCH. JIM, used by ORBIT and RENTER, produces $\vec{r}$ and $\vec{v}$ referenced to the true inertial frame. JOHN, a subroutine of both LAUNCH and RENTER, produces velocity components in an auxiliary system.

Stagnation heating rate is calculated for LAUNCH and RENTER by HEAT. The range from launch pad is computed for LAUNCH by the subroutine RFLP. ODOT presents ORBIT with the rates of change of the longitude of ascending node and of the argument of perigee.

5.2.1 IP 7090 Reference Frame Conversion Program (IPCNV)

This program, which is a subroutine to the LAUNCH program, converts IP 7090 position ($\vec{r}$) and velocity ($\vec{v}$) components (X, Y, Z), or processed data, to true inertial coordinates ($\underline{X}$, $\underline{Y}$, $\underline{Z}$). Inputs to IPCNV are placed into COMMON storage before entry. The program is entered with the call statement CALL IPCNV.

The conversion is performed in the following manner. If M is the transformation matrix below,

$$M = \begin{pmatrix} \cos(\nu_a + \omega_e t) & -\sin(\nu_a + \omega_e t) & 0 \\ \sin(\nu_a + \omega_e t) & \cos(\nu_a + \omega_e t) & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

then,

$$\vec{r} = \begin{pmatrix} \underline{X} \\ \underline{Y} \\ \underline{Z} \end{pmatrix} = M \cdot \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \qquad \vec{v} = \begin{matrix} \dot{\underline{X}} \\ \dot{\underline{Y}} \\ \dot{\underline{Z}} \end{matrix} = M \cdot \begin{pmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{pmatrix}$$

Prior to its exit, IPCNV places its output into COMMON storage.

5.2.2 B-GE Reference Frame Conversion Program (GECNV)

GECNV is a subroutine of the LAUNCH program. It converts processed data, position ($\vec{r}$) and velocity ($\vec{v}$) components, from the B-GE reference frame (ξ, η, ζ) to true inertial coordinates ($\overline{X}, \overline{Y}, \overline{Z}$). Inputs to GECNV are placed into COMMON storage before entry. The program is entered with the call statement CALL GECNV.

The conversion is performed in the following manner. If N is the transformation matrix below,

$$N = \begin{pmatrix} \cos(\delta\phi_R + \omega_e t) & -\sin(\delta\phi_R + \omega_e t) & 0 \\ \sin(\delta\phi_R + \omega_e t) & \cos(\delta\phi_R + \omega_e t) & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

then,

$$\vec{r} = \begin{pmatrix} \overline{X} \\ \overline{Y} \\ \overline{Z} \end{pmatrix} = N \cdot \begin{pmatrix} \xi \\ \eta \\ \zeta \end{pmatrix}$$

$$\vec{v} = \begin{pmatrix} \dot{\overline{X}} \\ \dot{\overline{Y}} \\ \dot{\overline{Z}} \end{pmatrix} = N \cdot \begin{pmatrix} \dot{\xi} \\ \dot{\eta} \\ \dot{\zeta} \end{pmatrix}$$

Prior to its exit, GECNV places its outputs into COMMON storage.

5.2.3 True Inertial Coordinate Conversion Program (MERCNV)

MERCNV, a subroutine of LAUNCH, converts position ($\vec{r}$) and velocity ($\vec{v}$) components in the true inertial coordinate frame ($\overline{X}, \overline{Y}, \overline{Z}$) to pad rectangular coordinates (u, v, w). All inputs quantities are placed into COMMON storage prior to entry with the call statement, CALL MERCNV.

The sequence of transformation for $\vec{r}$ is,

$$\begin{pmatrix} \xi_\rho \\ \eta_\rho \\ \zeta_\rho \end{pmatrix} = \begin{pmatrix} \cos(\delta\phi_p + \omega_e t) & \sin(\delta\phi_p + \omega_e t) & 0 \\ -\sin(\delta\phi_p + \omega_e t) & \cos(\delta\phi_p + \omega_e t) & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \overline{X} \\ \overline{Y} \\ \overline{Z} \end{pmatrix}$$

$$\begin{pmatrix} U' \\ V' \\ W' \end{pmatrix} = \begin{pmatrix} \eta_\rho \\ \zeta_\rho \\ \xi_\rho \end{pmatrix} - \begin{pmatrix} 0 \\ r'_0 \sin L_\rho \\ r'_0 \cos L_\rho \end{pmatrix}$$

$$\begin{pmatrix} \overline{U} \\ \overline{V} \\ \overline{W} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos L_p' & -\sin L_p' \\ 0 & \sin L_p' & \cos L_p' \end{pmatrix} \begin{pmatrix} U' \\ V' \\ W' \end{pmatrix}$$

$$\begin{pmatrix} u \\ v \\ w \end{pmatrix} = \begin{pmatrix} \sin\Theta_0 & \cos\Theta_0 & 0 \\ -\cos\Theta_0 & \sin\Theta_0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \overline{U} \\ \overline{V} \\ \overline{W} \end{pmatrix}$$

The sequence for $\vec{v}$ is:

$$\begin{pmatrix} \dot{\xi}_\rho \\ \dot{\eta}_\rho \\ \dot{\zeta}_\rho \end{pmatrix} = \begin{pmatrix} \cos(\delta\Theta_p + \omega_e t) & \sin(\delta\Theta_p + \omega_e t) & 0 \\ -\sin(\delta\Theta_p + \omega_e t) & \cos(\delta\Theta_p + \omega_e t) & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \dot{\underline{X}} + \omega_e \underline{Y} \\ \dot{\underline{Y}} - \omega_e \underline{X} \\ \dot{\underline{Z}} \end{pmatrix}$$

$$\begin{pmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{pmatrix} = \begin{pmatrix} \sin\Theta_0 & \cos\Theta_0 & 0 \\ -\cos\Theta_0 & \sin\Theta_0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos L_p' & -\sin L_\rho \\ 0 & \sin L_p' & \cos L_\rho \end{pmatrix} \begin{pmatrix} \dot{\eta}_\rho \\ \dot{\zeta}_\rho \\ \dot{\xi}_\rho \end{pmatrix}$$

MERCNV places its output values into COMMON storage prior to its exit.

5.2.4 Position and Velocity Processor Program (JIM)

JIM produces $\vec{r}$ and $\vec{v}$ referenced to true inertial frame. The input quantities, r_i, L_C, λ, γ_i, Ψ_i, and V, are placed into COMMON storage before entry to the program. JIM, which is entered by the call statement CALL JIM, is a subroutine of the programs ORBIT and RENTER.

The $\vec{r}$ and $\vec{v}$ transformations are written as follows:

$$\vec{r} = \begin{pmatrix} \underline{\overline{X}} \\ \underline{\overline{Y}} \\ \underline{\overline{Z}} \end{pmatrix} \doteq \begin{pmatrix} r_i \cos L_C \cos \lambda \\ r_i \cos L_C \sin \lambda \\ r_i \sin L_C \end{pmatrix}$$

$$\vec{v} = \begin{pmatrix} \dot{\underline{X}} \\ \dot{\underline{Y}} \\ \dot{\underline{Z}} \end{pmatrix} = \begin{pmatrix} \cos L_C \cos \lambda & -\cos \Psi_i \sin L_C \cos\lambda & \sin \Psi_i \sin\lambda \\ \cos L_C \cos\lambda & -\cos \Psi_i \sin L_C \sin\lambda & \sin \Psi_i \cos\lambda \\ \sin L_C & \cos \Psi_i \cos L_C & 0 \end{pmatrix} \begin{pmatrix} V_i \sin\gamma_i \\ V_i \cos\gamma_i \\ V_i \cos\gamma_i \end{pmatrix}$$

Prior to its exit, JIM places its output into COMMON storage.

5.2.5 Velocity Component Processor Program (JOHN)

This subroutine, used by both LAUNCH and RENTER, produces velocity components of the true inertial coordinate frame ($\dot{\underline{X}}$, $\dot{\underline{Y}}$, $\dot{\underline{Z}}$) in an auxiliary system ($\dot{x}$, $\dot{y}_i$, $\dot{z}$). Inputs to JOHN are in COMMON storage prior to entry with the call statement CALL JOHN.

The intermediate coordinate transformation performed by this program is effected by the following equation:

$$\begin{pmatrix} \dot{x} \\ \dot{y}_i \\ \dot{z} \end{pmatrix} = \begin{pmatrix} \cos L_C \cos \Delta\lambda_i & \cos L_C \sin \Delta\lambda_i & \sin L_C \\ -\sin \Delta\lambda_i & \cos \Delta\lambda_i & 0 \\ -\sin L_C \cos \Delta\lambda_i & -\sin L_C \sin \Delta\lambda_i & \cos L_C \end{pmatrix} \begin{pmatrix} \dot{\underline{X}} \\ \dot{\underline{Y}} \\ \dot{\underline{Z}} \end{pmatrix}$$

JOHN places its output into COMMON before it exits.

5.2.6 Atmospheric Density Processor Program (ATMOS)

This subroutine of LAUNCH and RENTER produces, in English units, the atmospheric density, kinematic viscosity and speed of sound at capsule position. ATMOS prepares for aerodynamic parameter calculations the mach number (ratio of relative speed to local speed of sound), Reynold's number per foot (ratio of relative speed to local kinematic viscosity), and dynamic pressure ($V_e^{\,2}/_2$).

Inputs to ATMOS, including the coefficients of density polynomials set up by ACTORS (see Subsection 3.2), are in COMMON storage before entry. Control is transferred to ATMOS with the call statement CALL ATMOS (KERR). The return indicator, KERR, is part of the output from the program. The remaining output is placed into COMMON before the program exits.

KERR = 2 when the height of the capsule exceeds 90 geopotential kilometers, in which case the speed of sound ceases to be uniquely defined independent of sonic frequency. Therefore, kinematic viscosity and the speed of sound (actually, the mach number and Reynold's number) are meaningless and are set to minus zero (-0). Alternatively, KERR = 1 if both kinematic viscosity and the speed of sound are calculated.

The equations used by ATMOS are given below:

$$h = .3048 * h \qquad \text{feet converted to meters}$$

$$h = \frac{a_e h}{a_e + h} \qquad \text{meters converted to geopotential meters}$$

The seventh degree polynomials of atmospheric density are:

$$L_n(\rho) = \sum_{i=0}^{7} a_i \left(\frac{h}{10^5}\right)^i \qquad h < 136025.0$$

$$= \sum_{i=0}^{7} b_i \left(\frac{h}{10^5}\right)^i \qquad h \geqq 136025.0$$

$$\rho = \exp\left[L_n(\rho)\right] \qquad (\text{kilograms/meters}^3)$$

$$T_M = (T_M)_b + L_M (h - h_b) \text{ for proper base value, b.}$$

$$C_S = 20.046333 \sqrt{T_M} \qquad (\text{meters/second})$$

$$\nu = \frac{1.458 \times 10^{-6} \times T_M^{3/2}}{(T_M + 110.4)} \qquad (\text{meters}^2/\text{second})$$

converting ν, C_S and ρ to English units:

$$\nu\ (\text{ft.}^2/\text{sec.}) = \nu/.9290304$$

$$C_S\ (\text{ft./sec.}) = C_S/.3048$$

$$\rho\ (\text{slug/ft.}^3) = \rho/515.378725$$

5.2.7 Stagnation Heat Rate Processor Program (HEAT)

HEAT calculates the stagnation heat rate, in English units, needed by LAUNCH and RENTER. Inputs to HEAT are placed into COMMON storage before entry. Transfer of control to the program is effected by the call statement, CALL HEAT. The program places its single output, q_s (BTU/sec. ft.^{2}), into COMMON before its exit.

The methodology involved in calculating these data is shown below:

If ρ_0 = sea level density (slugs/foot^3)

μ_∞ = free stream speed (feet/second, relative)

h_s = stagnation enthalpy (BTU/pound)

h_w = wall enthalpy (BTU/pound f(t)(p))

h_{w540} = reference wall enthalpy (=130 BTU/pound)

$\sqrt{R}$ = 2.58 (foot$^{1/2}$)

Then, in continuum flow (h $\leqq$ 350,000 feet)

$$q_s = \epsilon \sigma T^4, \text{ where } \epsilon = 0.8 \text{ and } \sigma = \frac{0.171 \times 10^8}{3600} \text{ BTU/hour}$$

and in free molecular flow (h > 350,000 feet)

$$q_s = 2.69 \times 10^7 \quad \eta \left(\frac{\rho_\infty}{\rho_o}\right) \left(\frac{\mu_\infty}{\mu_c}\right)^3 \text{ (BTU/feet}^2 \text{ per second)}$$

where ρ_∞ = free stream density (slugs/foot^3)

μ_c =capsule speed 26,000 (feet/second, inertial)

η =1.0

From these equations, then, the following can be stated:

$$q_s = 4.879 \sqrt{\rho} \quad U_D^{3.15}, \ h \leq 350,000 \text{ feet}$$

$$q_s = 6.439 \times 10^5 \rho \ U_D^{3}, \ h > 350,000 \text{ feet}$$

where U_D = free stream speed (kilofeet/second, earth-fixed)

ρ = free stream density (slug/foot^3)

q_s = stagnation heating rate (BTU/seconds per foot^2)

5.2.8 Range From Launch Pad Processor Program (RFLP)

RFLP, a LAUNCH program subroutine, calculates S, the range from the launch pad of the capsule. Inputs are placed in COMMON storage before the program is called with the statement CALL RFLP. This program uses the utility routine ARCCOS in calculating S, which is the only output of RFLP. S is placed into COMMON prior to the exit of RFLP. The equation for range from launch pad is as follows:

$$S = B \cdot \left(\frac{\overline{R}_p + R_L}{2} \right) \quad \text{where} \quad \frac{\overline{R}_p + R_L}{2} \text{ is an average radius}$$

and $B = \cos^{-1}(A)$. If $|A| > 1$, then S is set equal to zero and the program returns control to Monitor. If $|A| \leqq 1$, then S is calculated as follows:

$$S = \cos^{-1} (\sin L_D \sin L_p' \quad \cos L_D \cos L_p' \cos(\lambda - \lambda_1)) \cdot \left(\frac{\overline{R}_p + R_L}{2} \right)$$

before control returns to Monitor.

5.2.9 ω and Ω Processor Program (ODOT)

This subroutine of ORBIT calculates the derivatives of longitude of ascending node ($\dot{\Omega}$) and the argument of perigee ($\dot{\omega}$). Inputs are placed in COMMON storage prior to entering the program with the call statement CALL ODOT. The outputs are placed in COMMON before ODOT exits.

The equations for $\dot{\omega}$ and $\dot{\Omega}$ are as follows:

$$\dot{\omega} = 3.4722 \times 10^{-3} \left(\frac{a_e}{\rho}\right)^2 \left(\frac{a_e}{a}\right)^{3/2} (5 \cos^2 i - 1) \text{ (degrees/minute)}$$

$$\dot{\Omega} = -6.9444 \times 10^{-3} \left(\frac{a_e}{\rho} \right)^2 \left(\frac{a_e}{a} \right)^{3/2} \cos i \text{ (degrees/minute)}$$

where

a_e = radius of earth

ρ = semi-latus rectum

a = semi-major axis

i = inclination angle

The calculations presented above are based on values obtained from earlier satellite data reduction and the general formulas,

$$\dot{\Omega} = \frac{-2\pi}{P} \cdot \frac{\cos i}{p^2} \cdot J_2 - \frac{2\pi}{P} \cdot \frac{\cos i}{p^4} \left(1+\frac{3e^2}{2}\right)\left(1-\frac{7}{4}\sin^2 i\right) \cdot J_4$$

$$\dot{\omega} = -\dot{\Omega}\cos i + \frac{2\pi}{P}\left(\frac{1-\frac{3}{2}\sin^2 i}{p^2}\right) J_2 + \frac{2\pi}{P}\,\frac{\left(1+\frac{3}{4}e^2\right)\left(1-5\sin^2 i+8\sin^4 i\right)}{p^4} J_4$$

where P = period, p = latus rectum of orbit, and J_2, J_4 are derived from the gravitational potential expansion in Legendre polynomial functions of declination (i.e., geocentric latitude).

SECTION 6
UTILITY PROGRAMS

The program has seven subroutines that are of a utility nature. These programs are used so frequently in Monitor and other Postflight Reporter programs that they merit separate definition. One routine (HRSCNV) is concerned with the conversion of time. Three of them (FIXIT, FIXIT1 and FIXIT2) involve the fixing of hours and angles within specified ranges. The remaining three (TAN, ARCSIN and ARCCOS) are arithmetic and produce the tangent, arc-sine and arc-cosine functions.

6.1 TIME CONVERSION PROGRAM (HRSCNV)

HRSCNV converts floating point time (seconds) into fixed point hours, minutes and seconds whenever called upon in the various subprograms. The input parameter, TIME, and output parameters, NHOUR, MIN, and NSEC, are contained in the call statement CALL HRSCNV (TIME, NHOUR, MIN, NSEC). The output, integral hours, minutes and seconds, is placed into the parameter list of the call statement.

6.2 ANGLE DEGREE LIMITS DETERMINATION PROGRAM (FIXIT)

This program ensures that an angle lies between zero and 360 degrees (including zero degrees). The angle is specified in degrees in the call statement CALL FIXIT (A). If the angle does not lie between zero and 360 degrees, FIXIT modifies it by 360 degree increments until it is between the two limits. The modified angle then replaces (A), the parameter in the call statement.

6.3 ANGLE RADIAN LIMITS DETERMINATION PROGRAM (FIXIT1)

FIXIT1 ensures that an angle lies between zero and two pi (2π) radians (including zero radians). The angle is specified in radians in the call statement CALL FIXIT1 (A). If the angle does not lie between zero and two pi radians, FIXIT1 modifies it by two pi radian increments until it is between the two limits. The modified angle then replaces (A), the parameter in the call statement.

6.4 HOUR LIMITS DETERMINATION PROGRAM (FIXIT2)

FIXIT2 ensures that N hours lie between the limits of zero and 24 (including zero). The number of hours, N, is specified in the call statement CALL FIXIT2

(N). If N does not lie between zero and 24, FIXIT2 modifies it by 24-hour increments until it is between the two limits. The modified N then replaces (N), the parameter in the call statement.

6.5 TANGENT COMPUTATION PROGRAM (TAN)

TAN derives a function whose value is the trigonometric tangent of the parameter ANGLE. First, the angle specified in the call statement Y = TAN (ANGLE) is determined to be between zero and two pi radians (FIXIT1 ensures this). Then, TAN calculates the tangent of the corresponding first quadrant (acute) angle. The sign is determined by quadrant considerations. For infinite values, the largest possible algebraic value is set equal to the function.

6.6 ARC-SINE COMPUTATION PROGRAM (ARCSIN)

ARCSIN produces the radian angle between $-\pi/2$ and $+\pi/2$ whose sine corresponds to the value given in the call statement ANGLE = ARCSIN (VALUE). However, if VALUE is numerically less than +1.0, the program uses the FORTRAN routine ATAN. If the argument exceeds + 1.0 numerically, $\pm\frac{\pi}{2}$ is returned.

6.7 ARC-COSINE COMPUTATION PROGRAM (ARCCOS)

ARCCOS produces the radian angle between zero and π radians whose cosine corresponds to the value given in the call statement ANGLE = ARCCOS (VALUE). If VALUE is between minus and plus one (-1, +1), a transformation using ARCSIN is accomplished. If VALUE is greater than one (>1), zero is returned. If VALUE is less than or equal to minus one (≤ -1), π is returned.

SECTION 7
PROGRAM OPERATING PROCEDURES

The computer operator should use the following specific procedures to execute the Postflight Reporter Program properly. They include details on tape set-up, entry keys, card reader status, sense switch settings, and program stops. The composition of the data deck is specified for the user's information.

7.1 SPECIFIC PROCEDURES

Tape Set-Up

Mount the following tapes:

A1 - 32K 709/7090 FORTRAN System tape (complete set-up)

A4 - High-Speed Output Message tape

A5 - Stand-by to A4

B1 - Miscellaneous data tape—accepts bad data from the log tape(s) and data rejected from the programs.

B4 - High-Speed Input Message tape—first file contains high-speed input data; second file contains 1,000 records of the 7 discrete events for GETME.

B6 - Mercury Program System Log tape

B7 - Stand-by to B6

C3 - Output tape for the postflight report

Entry Keys

The number of physical log tapes is right-justified in the address portion of the entry keys.

On-Line Card Reader

Data deck in and ready.

Sense Switch Settings

SS1 UP - suppress on-line copy of C3
DOWN - print C3 on-line

SS2 UP - final report on C3
DOWN - multiple reports on C3

SS6 UP - execute SORTER
DOWN - suppress SORTER

Program Stops

HPR 77771_8 - on-line printout, PLEASE CHECK INSTRUCTIONS TO DETERMINE SETTING OF SENSE SWITCH 6. SET AND PRESS START.

HPR 77772_8 - on-line printout when SORTER is suppressed and sorted tapes are available, PLEASE CHECK TO SEE IF ALL TAPE UNITS ARE SET PROPERLY. IF SO, PLEASE PRESS START.

HPR 77773_8 - error stop from GETME, on-line printout, AN ERROR HAS BEEN DETECTED IN KEY OR TAPE SET-UP. PLEASE REVIEW OPERATING NOTES. AFTER COMPLETING CORRECTION PRESS START.

HPR 77774_8 - on-line printout for SORTER execution, PLEASE CHECK TO SEE THAT NUMBER OF PHYSICAL LOG TAPES IS ENTERED IN ADDRESS PORTION OF KEYS.

HPR 77775_8 - error stop from SORTER, same on-line printout as for 77773_8.

HPR 77776_8 - end of job has been reached, EOF is written on C3.

HPR 77777_8 - final stop or second report; print C3 under program control at 6 lines per inch.

HPR 12121_8 - error stop for request-for-report card, on-line printout, ERROR IN DECK SET-UP. TF EXCEEDS TL. REPUNCH, PRESS START.

7.2 DATA DECK COMPOSITION

Card	Card Columns	Digital Formats	Data
1	1-72 (inclusive)		Headings for each phase (see Appendix C)
2	1-9	XXXXXXX.X	Canonical equatorial radius (EQURAD)
	11-22	X.XXXXXXXXXX	Earth flattening (FFLAT)
	24-32	X.XXXXXXX	Gravity (GRAVIT)
	34-42	X.XXXXXXX	Earth's gravitational constant (XMUE)
	44-52	XXXX.XXXX	2nd harmonic potential (CANJ2)
	54-62	X.XXXXXXX	3rd harmonic potential (CANJ3)
	64-72	X.XXXXXXX	4th harmonic potential (CANJ4)
3	1-10	XXXXXXXX.X	Clarke spheroid equatorial radius (AC)
	12-21	XXXXXXXX.X	Polar radius (BC)
	23-32	XXXXXXXX.X	Radial distance of spherical earth (RBAR)
	34-48	X.XXXXXXXXXXXXX	Angular rotation of earth (OMEGAE)
4	1-10	XX.XXXXXXX	Geodetic latitude of pad (XLPAD)
	12-20	XX.XXXXXX	Longitude of pad (XLM1)
	22-31	+XXXXX.XXX	Height of pad above mean sea level (HPAD)
	33-41	XX.XXXXXX	Longitude of GE central radar (XLMO)
	43-51	XXX.XXXXX	Greenwich hour angle at GMTLO (XNUA)

DATA DECK COMPOSITION (Cont'd)

Card	Card Columns	Digital Formats	Data
	53-62	XXX.XXXXX	Launch azimuth (THETA0)
5 (to last)	1	X	1 = Three-Day Report; 2 Quick Look
	2-3 4-5 6-9	XX XX XX.X	Integral hours/minutes/seconds = TF = first time since phase initiation
	10-11 12-13 14-17	XX XX XX.X	Integral hours/minutes/seconds = TF = last time since phase initiation
	18-22	XXX.X	Time intercal (Δt) between report output
	23 22	X	Phase indicator 1 = Launch 2 = Abort, 3 = Orbit, 4 = Reentry
Last	1		Sign-off. Zero punch in column 1

APPENDIX A

POSTFLIGHT REPORTER SYMBOLIC DESIGNATIONS

This appendix includes both the FORTRAN and the mathematical symbology used to produce the postflight reports. The material in the FORTRAN list is arranged alphabetically. Coordinate systems appearing at the end of this appendix are defined under the appropriate heading in Appendix B, with the exception of the last one—the intermediate rotational coordinate system which is covered in NASA Working Paper 146.

FORTRAN Symbolic Names	Mathematical Symbology	Definition
AC	a_c	Equatorial radius, Clarke 1866 Spheroid
ARBAR	$a - \bar{R}$	Semi-major axis less spherical radius
AREA	Area	Selected recovery area
ARGP	ω	Argument of perigee (defined in volume MC 102)
ARGP1	$\dot{\omega}$	Angular rotation of ω
ASUBR	A_R	Resultant acceleration—radial component of the instantaneous acceleration, g units
AXIS	a	Semi-major axis
BC	b_c	Polar radius, Clarke 1866 Spheriod
CANJ2	J_2	Canonical 2nd harmonic potential
CANJ3	J_3	Canonical 3rd harmonic potential
CANJ4	J_4	Canonical 4th harmonic potential
CL	L_C	Geocentric latitude

FORTRAN Symbolic Names	Mathematical Symbology	Definition
CR	$y - y_{nominal}$	Crossrange distance
CS	C_S	Speed of sound
D	d	Downrange distance
DENS	ρ	Atmospheric density
DL	L_D	Geodetic latitude—latitude of present position
DLMI	$\Delta \lambda_i$	Angle in equatorical plane between X and the projection of capsule into that plane
DPHIR	$\delta \theta_R$	Angle at 2" lift-off in the equatorial plane between Υ and longitude of GE radar
DTR	Δt_r	Time delay to retrofire
DTHTAP	$\delta \theta_p$	Angle in equatorial plane between Υ and longitude of pad
EA	E	Eccentric anomaly, degrees
ECC	e	Eccentricity (defined in Volume MC 102)
ECTRC	ECTRC	Elapsed capsule clock time of retro-fire computed
ECTRC1	$ECTRC_1$	ECTRC for emergency recovery area
ECTRC2	$ECTRC_2$	ECTRC for end of present orbit
ECTRC3	$ECTRC_3$	ECTRC for end of normal 3-orbit mission
ECTRS	ECTRS	Elapsed capsule time to retrofire, presently set
EGT	EGT	Elapsed ground time since retrofire occurred

FORTRAN Symbolic Names	Mathematical Symbology	Definition
EQURAD	a_e	Canonical equatorial radius
FFLAT	f	Canonical flattening
GE	γ_e	Earth-fixed flight-path angle
GI	γ_i	Inertial flight-path angle (positive above local horizon), degrees
GIGE	$(\gamma_i - \gamma_{nom})GE$	Actual minus nominal flight-path angle for B-GE data, degrees
GIIP	$(\gamma_i - \gamma_{nom})IP$	For IP-7090 data actual minus nominal flight-path angle for IP 7090 data, degrees
GMTLC	GMTLC	Greenwich mean time of landing
GMTLO	GMTLO	Greenwich mean time of 2'' lift-off
GMTRC	GMTRC	Greenwich mean time of retrofire computed
GMTRC1	$GMTRC_1$	GMTRC for emergency recovery area
GMTRC2	$GMTRC_2$	GMTRC for end of present orbit
GMTRC3	$GMTRC_3$	GMTRC for end of normal 3-orbit mission
GMTRS	GMTRS	Present capsule retrofire clock setting computed
GRAVIT	$\dot{g}$ e	Canonical value of gravity at equator
GTL	GTL	Time until landing
GTRS	GTRS	Time left until retrofire will occur
HA	h_a	Apogee height
HE	h_e	Height above oblate earth
HPAD	h_o	Height of pad above mean sea-level

FORTRAN Symbolic Names	Mathematical Symbology	Definition
OME	Ω	Longitude of ascending node (defined in volume MC 102)
OME1	$\dot{\Omega}$	Angular rotation of Ω
OMEGAE	ω_e	Angular rotation of earth
P	p	Semi-latus rectum
PER	T	Orbital period, minutes
PHI	ϕ	Total angle in plane from longitude of node to capsule
PHIIP	ϕ_{IP}	Geodetic latitude of refired impact point
PHIMAX	ϕ_{max}	Geodetic latitude - maximum time delay - impact point, degrees
PHIMIN	ϕ_{min}	Geodetic latitude - minimum time delay - impact point, degrees
PSIE	ψ_e	Earth-fixed heading
PSII	ψ_i	Inertial heading
QD	q_D	Dynamic pressure
QS	q_s	Heating rate, BTU/ft^2 second
RADIUS	r	Geocentric radial distance to the capsule, feet
RBAR	$\bar{R}$	Radial distance of a spherical earth
RL	r_l	Local radius of earth (i.e., at the geocentric latitude of capsule)
RN	R_N	Reynolds number
RPBAR	$\bar{R}_p$	Geocentric radius at pad
RRBAR	$r - \bar{R}$	Height above spherical earth, nautical miles, $\bar{R}$ = 20,910,000 feet

FORTRAN Symbolic Names	Mathematical Symbology	Definition
R01	r_o	$(=\bar{R}_p + h_o=)$ Total distance from geocenter to pad
S	S	Range from launch pad
SR	S_R	Re-entry range from retrofire, nautical miles
T	t	Elapsed time since lift-off
TA	θ_1	True anomaly
THETA0	θ_0	Launch azimuth, from north, positive clockwise
TP	t_p	Elapsed time from perigee passage
VE	V_e	Earth-fixed velocity, ft/sec.
VI	V_i	Inertial velocity, ft/sec.
VIVR	V_i/V_R	Speed ratio
VIVRGE	$(V_i/V_R - V_i/V_R^{nom})GE$	Actual minus nominal velocity ratio for B-GE data
VIVRIP	$(V_i/V_R - V_i/V_R^{nom})IP$	Actual minus nominal velocity rate for IP 7090 data
XICTRC	ICTRC	Incremental change to reset clock
XINC	i	Inclination angle (defined in volume MC 102)
XL	λ	Earth-fixed longitude
XLAMP	λ_p	Longitude of perigee at perigee passage
XLMAX	ϕ_{max}	Longitude - maximum time delay - impact point, degrees
XLM1	λ_1	Longitude of launch pad

FORTRAN Symbolic Names	Mathematical Symbology	Definition
XLMMIN	λ_{min}	Longitude - minimum time delay - impact point, degrees
XLMIP	λ_{IP}	Longitude of refired impact point
XLPAD	L_P'	Geodetic latitude of pad
XLRHO	L_ρ	Geocentric latitude of pad
XMA	M_a	Mean anomaly
XMACH	M	Mach number
XMUE	μ_e	Earth's gravitational constant (0 th order harmonic)
NORBCP	NORBCP	Orbit capability
XNU	ν	Kinematic viscosity
XNUA	ν_a	Greenwich hour angle from Υ at midnight preceding launch
NUMORB	NUMORB	Orbit number
X	$\underline{X}$	
Y	$\underline{Y}$	
Z	$\underline{Z}$	True Inertial coordinate system (See Appendix B)
X1	$\dot{\underline{X}}$	
Y1	$\dot{\underline{Y}}$	
Z1	$\dot{\underline{Z}}$	
U	u	
V	v	
W	w	Pad coordinate system (See Appendix B)
U1	$\dot{u}$	
V1	$\dot{v}$	
W1	$\dot{w}$	

FORTRAN Symbolic Names	Mathematical Symbology	Definition
XIP	X	IP 7090 coordinate system (See Appendix B)
YIP	Y	
ZIP	Z	
XIP1	$\dot{X}$	
YIP1	$\dot{Y}$	
ZIP1	$\dot{Z}$	
XI	ξ	B-GE coordinate system (i.e., GE-quasi-inertial) (See Appendix B)
ETA	η	
ZETA	ζ	
XI1	$\dot{\xi}$	
ETA1	$\dot{\eta}$	
ZETA1	$\dot{\zeta}$	
YI1	$\dot{y}_i$	Intermediate rotational coordinate system (See NASA Working Paper 146)
XX	x	
YY	y	
ZZ	z	
XX1	$\dot{x}$	
YY1	$\dot{y}$	
ZZ1	$\dot{z}$	

APPENDIX B

COORDINATE CONVERSION SYSTEM

This appendix deals with the following coordinate systems involved in the Postflight Reporter Program:

True inertial coordinates	$\underline{\bar{X}}, \underline{\bar{Y}}, \underline{\bar{Z}}, \dot{\underline{X}}, \dot{\underline{Y}}, \dot{\underline{Z}}$
GE quasi-inertial coordinates	$\xi, \eta, \zeta, \dot{\xi}, \dot{\eta}, \dot{\zeta},$
IP 7090 quasi-inertial coordinates	$X, Y, Z, \dot{X}, \dot{Y}, \dot{Z}$
Pad rectangular coordinates	$u, v, w, \dot{u}, \dot{v}, \dot{w}$

In addition, the appendix includes definitions of Miscellaneous Transformations.

TRUE INERTIAL COORDINATE SYSTEM ($\underline{\bar{X}}, \underline{\bar{Y}}, \underline{\bar{Z}}, \dot{\underline{X}}, \dot{\underline{Y}}, \dot{\underline{Z}}$)

This fundamental system is a right-handed rectangular system centered at the earth's center, $\underline{\bar{X}}$ pointing to the first point of Aries (Υ) or vernal equinox, $\underline{\bar{Z}}$ lying along the earth's polar axis, and $\underline{\bar{Y}}$ normal to the meridian plane. In general, data in other systems are transformed to these coordinates prior to processing. In the Postflight Reporter program, IPCNV converts IP 7090 data to true inertial reference and GECNV converts B-GE data to true inertial reference. However, MERCNV converts true inertial coordinates to pad rectangular coordinates.

GE QUASI-INERTIAL COORDINATE SYSTEM ($\xi, \eta, \zeta, \dot{\xi}, \dot{\eta}, \dot{\zeta}$)

At two-inch lift-off, this system has the following configuration: ξ and η are axes lying in the earth's equatorial plane, with ξ lying in the meridian plane of the GE central radar, ξ is the normal lying along the earth's polar axis. The orlgin is at the earth's center and this system is a rectangular right-handed set of axes.

If $\delta \phi_R$ is the angle between the meridian plane through the first point of Aries (Υ) and the ξ, ζ plane at two-inch lift-off, then with terrestrial polar rotation rate of ω_e radians/second after t seconds, this angle is changed by

$\omega_e t$ to the value $\delta\phi_R + \omega_e t$. Therefore, if ν_a is the Greenwich mean hour angle at two-inch lift-off and if λ_0 represents the longitude of the GE central radar, then $\delta\phi_R = \nu_a + \lambda_0$ (actually, λ_0 is negative for Western longitudes).

Since the relationship between the GE and true inertial frames is merely a rotation about the common $\underline{Z}$ or ζ axis of angle $\delta\phi_R + \omega_e t$, the transformation may be written.

$$\begin{pmatrix} \underline{X} \\ \underline{Y} \\ \underline{Z} \end{pmatrix} = \begin{pmatrix} \cos(\delta\phi_R + \omega_e t) & -\sin(\delta\phi_R + \omega_e t) & 0 \\ \sin(\delta\phi_R + \omega_e t) & \cos(\delta\phi_R + \omega_e t) & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \xi \\ \eta \\ \zeta \end{pmatrix}$$

Thus, $\vec{r}_{true} = M\vec{r}_{GE}$ where M is the above matrix. The velocity relationship is obtained by differentiation, $\vec{V}_{true} = M\,\vec{V}_{GE}$, or explicitly

$$\begin{pmatrix} \dot{\underline{X}} \\ \dot{\underline{Y}} \\ \dot{\underline{Z}} \end{pmatrix} = \begin{pmatrix} \cos(\delta\phi_R + \omega_e t) & -\sin(\delta\phi_R + \omega_e t) & 0 \\ \sin(\delta\phi_R + \omega_e t) & \cos(\delta\phi_R + \omega_e t) & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \dot{\xi} \\ \dot{\eta} \\ \dot{\zeta} \end{pmatrix}$$

IP 7090 QUASI-INERTIAL COORDINATE SYSTEM ($X, Y, Z, \dot{X}, \dot{Y}, \dot{Z}$)

At two-inch lift-off, this system has the following configuration: X lies in the earth's meridian plane through Greenwich and in the equatorial plane. Z lies in the polar axis. Y lies in the equatorial plane normal to the X, Z plane and such that X, Y, Z forms a right-handed system.

If ν_a is the Greenwhich hour angle at two-inch lift-off, then with terrestrial angular rotation rate about the polar axis of ω_e radians/second, after t seconds this angle is changed by $\omega_e t$ to the value $\nu_a + \omega_e t$.

The relation between IP and true inertial coordinates is analagous to the GE and true inertial relationship. The conversion between systems is given by the following transformation:

$$\begin{pmatrix} \underline{X} \\ \underline{Y} \\ \underline{Z} \end{pmatrix} = \begin{pmatrix} \cos(\nu_a + \omega_e t) & -\sin(\nu_a + \omega_e t) & 0 \\ \sin(\nu_a + \omega_e t) & \cos(\nu \;+ \omega_e t) & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix}$$

$$\begin{pmatrix} \dot{\underline{X}} \\ \dot{\underline{Y}} \\ \dot{\underline{Z}} \end{pmatrix} = \begin{pmatrix} \cos(\nu_a + \omega_e t) & -\sin(\nu_a + \omega_e t) & 0 \\ \sin(\nu_a + \omega_e t) & \cos(\nu_a + \omega_e t) & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{pmatrix}$$

PAD RECTANGULAR COORDINATE SYSTEM $(u, v, w, \dot{u}, \dot{v}, \dot{w})$

This is a rectangular coordinate system with origin at the launch pad. The u axis points downrange, the w axis is normal to the tangent plane at the pad and directed away from the surface, and the v axis is normal to the u, w plane and directed eastward.

Since the system is hinged at the surface, a combination of rotations and translations is necessary to convert from true inertial to the pad rectangular system. If $\delta\theta_p$ is the hour angle (at two-inch lift-off) of the launch pad, then $\delta\theta_p = \nu_a + \lambda_{PAD}$ ($\lambda_{PAD} < 0$ for western longitudes). Then, as in previous cases, after t seconds, $\vec{r}_{rot} = M\vec{r}_{inertial}$ is used to pass from true inertial to rotational. To translate from geocenter to pad, $\vec{r}_{rot} = M\vec{r}_{inertial} + \vec{r}_{0'}$.

In the tangent plane, rotation puts $\vec{r}_{rot}$ relative to the equatorial plane—normal $\vec{r}_{rot'} = N\vec{r}_{rot}$ through L_ρ (geocentric latitude of the pad). Finally, to rotate from east-north in tangent plane to downrange vs. crossrange through the angle θ_0 (planned azimuth) $\vec{r}_{pad} = L\vec{r}_{rot'}$. The matrices for the transformations are:

$$M = \begin{pmatrix} \cos(\delta\theta_p + \omega_e t) & \sin(\delta\theta_p + \omega_e t) & 0 \\ -\sin(\delta\theta_p + \omega_e t) & \cos(\delta\theta_p + \omega_e t) & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$\vec{r}_{0'} = -\begin{pmatrix} r_{0'} \cos L_\rho \\ 0 \\ r_{0'} \sin L_\rho \end{pmatrix} \qquad \text{where } r_{0'} = \bar{R}_p + h_0$$

for geocentric pad latitude L_ρ $\left(= \tan^{-1}\left[\left(\frac{b_c}{a_c}\right)^2 \tan L_{p'}\right]\right.$, $L_{p'} =$ geodetic).

$$\bar{R}_p = \frac{a_c}{\sqrt{\cos^2 L_\rho + \left(\frac{a_c}{b_c}\right)^2 \sin^2 L_\rho}}$$

h_0 = height of the pad above mean sea level

$$N = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos L_{p'} & -\sin L_{p'} \\ 0 & \sin L_{p'} & \cos L_{p'} \end{pmatrix}$$

$$L = \begin{pmatrix} \sin \theta_0 & \cos \theta_0 & 0 \\ -\cos \theta_0 & \sin \theta_0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

To obtain the velocity relation, the translation and rotation matrices are compounded and differentiated to find,

$$\begin{pmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{pmatrix} = \begin{pmatrix} \sin \theta_0 & \cos \theta_0 & 0 \\ -\cos \theta_0 & \sin \theta_0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos L_{p'} & -\sin L_{p'} \\ 0 & \sin L_{p'} & \cos L_{p'} \end{pmatrix}$$

$$\begin{pmatrix} \cos(\delta\theta_p + \omega_e t) & \sin(\delta\theta_p + \omega_e t) & 0 \\ -\sin(\delta\theta_p + \omega_e t) & \cos(\delta\theta_p + \omega_e t) & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \dot{\bar{X}} + \omega_e \bar{Y} \\ \dot{\bar{Y}} - \omega_e \bar{X} \\ \dot{\bar{Z}} \end{pmatrix}$$

MISCELLANEOUS TRANSFORMATIONS

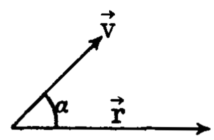
To obtain the geocentric latitude L_C from the geodetic latitude L_D with respect to a spheroid of equatorial radius, a, polar radius, b, the relation $\tan L_C = \left(\frac{b}{a}\right)^2 \tan L_D$ is used. To find the geocentric distance $\bar{R}_C$ to a point on the spheroid whose geocentric latitude is L_C, $\bar{R}_C = \dfrac{a}{\sqrt{\cos^2 L_C + \left(\frac{a}{b}\right)^2 \sin^2 L_C}}$ is used.

Position in geocentric spherical coordinates (r, L_C, λ) may be expressed in (geocentric) true inertial frame by the relationships $\underline{\bar{X}} = r \cos L_C \cos \lambda$, $\underline{\bar{Y}} = r \cos L_C \sin \lambda$, $\underline{\bar{Z}} = r \sin L_C$.

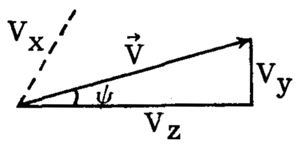
Derivations for data used in the Postflight Reporter processing programs and their subroutines are shown below.

Flight path angle, γ

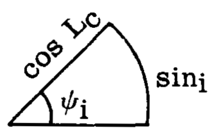
$$\vec{r} \cdot \vec{v} = rv \cos \alpha \qquad \text{but } \alpha = 90° - \gamma$$
$$= rv \sin \gamma$$



Heading angle, ψ



Project $\bar{V}$ onto YZ plane, $\tan \psi = \dfrac{V_y}{V_z}$



In a spherical triangle, $\sin \psi_i = \dfrac{\sin_i}{\cos L_C}$

Mach number, M

This is defined as the relative speed - local speed of sound ratio.

$M = V_e / C_S$, where $|\vec{V}_e| = |\vec{V}_i - \vec{\omega}_e \times \vec{r}|$ is the speed with respect to rotating atmosphere.

Reynolds number (per foot)

This is defined as the ratio of relative speed to local kinematic viscosity.

$R_N = V_e / \nu$.

Dynamic pressure

Dynamic pressure is the ratio of drag to drag coefficient. Therefore, $q_D = 1/2 \rho \ V_e 2$, where ρ is the local atmospheric density.

Stagnation heating rate

This is a function of altitude - for free stream $q_1 = K\rho \ u^3$

for $q = K_1 \sqrt{\rho} \ u^{3.15}$

where K, K_1 are given and u is free stream speed.

Downrange distance (RFLP)

The formula for downrange distance is the spherical trigonometric relation (from the spherical triangle), $\theta = \cos^{-1} \left[\sin L_D \sin L_\rho + \cos L_D \cos L_\rho \cos (\lambda - \lambda_p) \right] \left[\frac{\bar{R}_p + \bar{R}}{2} \right]$.

APPENDIX C
REPORT DATA FORMATS

This appendix presents the data formats for each of the two postflight reports, the Quick Look and the Three Day Report. In accordance with specifications of NASA-Space Task Group the formats and heading comprise the key for transmission of postflight data. Each report presents, in a specific time interval, selected data about each applicable phase of the mission, as follows:

Phase	Quick Look	Three-Day Report
Launch	4 seconds	0.5 seconds
Abort	4 seconds	1.0 seconds
Orbit	2 minutes	24.0 seconds
Re-entry	12 seconds	6.0 seconds

At the beginning of each phase, a heading is printed, giving the name of the phase, the GMT of lift-off, and the longitude of Aries at lift-off, in degrees. After each heading, a paragraph containing postflight information about the phase under investigation is written in fixed-point format. The data formats for each report are shown below. (The mathematical symbology, which is defined in Appendix A, represents digital data from the Postflight Reporter Program.)

QUICK LOOK DATA FORMATS

Launch

Line	1	t, sec	u	v	w	$\dot{u}$	$\dot{v}$	$\dot{w}$
	2		X	Y	Z	$\dot{X}$	$\dot{Y}$	$\dot{Z}$
	3		V_i	γ_i	ψ_i	L_C	λ	h_e
	4	t, hrs	V_e	γ_e	ψ_e	L_D	r	$(r - \overline{R})$
	5	t, min	d	y	V/V_R	A_R	S	
	6	t, sec	M	q_D	R_N	q_S		

Abort

Line 1	t, sec	u	v	w	$\dot{u}$	$\dot{v}$	$\dot{w}$
2		X	Y	Z	$\dot{X}$	$\dot{Y}$	$\dot{Z}$
3		V_i	γ_i	ψ_i	L_C	λ	h_e
4	t, hrs	V_e	γ_e	ψ_e	L_D	r	$(r - \bar{R})$
5	t, min	d	y	V/V_R	A_R	S	
6	t, sec	M	q_D	R_N	q_S		

Orbit

Line 1	t, min	X	Y	Z	$\dot{X}$	$\dot{Y}$	$\dot{Z}$
2		V_i	γ_i	ψ_i	L_C	λ	h_e
3		V_e	γ_e	ψ_e	L_D	r	$(r - \bar{R})$
4		Blank					
5		t_p	T	e	i	ω	$\dot{\omega}$
6		$a - \bar{R}$	θ_1	E	M	Ω	$\dot{\Omega}$

Re-entry

Line 1	t, min	X	Y	Z	$\dot{X}$	$\dot{Y}$	$\dot{Z}$
2		V_i	γ_i	ψ_i	L_C	γ	h_e
3		V_e	γ_e	ψ_e	L_D	r	$(r - \bar{R})$
4		M	q_D	S_R	q_S		

THREE-DAY REPORT DATA FORMATS

Launch

Lines 1 through 6 are identical to the Launch format for the Quick Look.

Line 7 ϕ_{min} λ_{min} ϕ_{max} λ_{max} i Area

8 GO-NOGO Δt_R GMTRC

9 $(V/V_R - V/V_R^{nom})GE$ $(\gamma - \gamma_{nom})GE$ $(V/V_R - V/V_R^{nom})IP$

$(\gamma - \gamma_{nom})IP$

Abort

Lines 1 through 6 are identical to the Abort format for the Quick Look.

Line 7 GMTRC ECTRC ICTRC GMTLC

8 Area ϕ_{IP} λ_{IP}

Orbit

Lines 1 through 6 are identical to the Orbit format for the Quick Look.

Line 7 Orbit number Orbit capability h_a Area ϕ_{IP} λ_{IP}

8 $(GMTRC)_1$ $(ECTRC)_1$ $(GMTRC)_2$ $(ECTRC)_2$ $(GMTRC)_3$ $(ECTRC)_3$

9 GMTRS ICTRC GTRS GMTLC λ_p

Re-entry

Lines 1 through 4 are identical to the Re-entry format for the Quick Look.

Line 5 Area ϕ_{IP} λ_{IP} EGT GTL GMTLC